

**Vysoká škola báňská – Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra telekomunikační techniky**

**Praktická implementace sjednocené komunikace v  
podnikovém prostředí**

**Practical implementation of unified communication in  
enterprise environment**

**2014**

**Bc. Lukáš Orčík**

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra telekomunikační techniky

## Zadání diplomové práce

Student:

**Bc. Lukáš Orčík**

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2601T013 Telekomunikační technika

Téma:

**Praktická implementace sjednocené komunikace v podnikovém prostředí**  
**Practical Implementation of Unified Communication in Enterprise Environment**

Zásady pro vypracování:

Unified messaging (UM) umožňuje předání zprávy uživateli pomocí různých komunikačních kanálů. Tyto kanály si nastavuje sám uživatel dle preferencí, např. hlasová zpráva je přijatá e-mailem anebo přijatá e-mailová zpráva je reprodukována pomocí TTS (Text-to-Speech) modulu jako hlasová. Cílem diplomové práce je praktická realizace takového řešení v podnikovém prostředí.

1. Úvod do problematiky sjednocené komunikace UC (Unified Communications).
2. Komponenty UC pro služby TTS, presence, telefonie, hlasové a elektronické pošty.
3. Návrh a realizace platformy integrující otevřená řešení UC v prostředí GNU/Linux.
4. Zhodnocení dosažených výsledků.

Seznam doporučené odborné literatury:

1. M. Vozňák, Voice over IP. Vysokoškolská skripta, Vydavatel: VŠB-TU Ostrava, Dotisk prvního vydání, v Ostravě, 2009, ISBN 978-80-248-1828-3.
2. M. Voznak, K. Tomala, J. Vychodil, J. Slachta, Advanced concept of voice communication server on embedded platform, Przegląd Elektrotechniczny, Volume 89, Issue 2 B, 2013, pp. 228-233.
3. L. Macura, M. Voznak, J. Slachta, Unified Administration of VoIP Communication Systems, In Proc. International Conference on Telecommunication Systems, Modeling and Analysis (ICTSM2012), Prague, May 24-26, 2012, pp. 188-197, ISBN 978-0-9820958-6-7.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Miroslav Vozňák, Ph.D.**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014

doc. Ing. Miroslav Vozňák, Ph.D.  
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## Prehlásenie študenta

Prehlasujem, že som tuto diplomovou práci vypracoval samostatne. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

V Ostrave dna: *6.mája 2014*



.....  
podpis študenta

## **Pod'akovanie**

Rád by som poďakoval doc. Ing. Miroslavovi Vozňákovi, Ph.D. za odbornou pomoc a konzultácie pri vytvorení tejto diplomové práce.

## **Abstrakt**

Táto diplomová práca sa zaoberá problematikou zjednotenej komunikácie. Sú tu popísané rôzne problémy a následky ktoré vznikajú pri používaní viacerých druhov komunikačných kanálov. Povieme si niečo o najčastejšie používaných protokoloch pre komunikáciu. Vytvoríme návrh platformy, ktorá bude zjednocovať požadované protokoly do jednotného systému. V tomto návrhu sa pozrieme bližšie na problémy, ktoré tu vznikajú a budeme navrhovať riešenia, ktoré tieto problémy eliminujú. Definujeme si požiadavky, ktoré musí spĺňať systém, ktorý bude riešiť problematiku UC. Pomocou vytvorených návrhov implementuje mechanizmy pre jadro systému a jednotlivé moduly. Vytvoríme webové používateľské rozhranie a integrujeme doňho služby ako je napríklad preklad textu na hlas. Ďalej v tomto rozhraní vytvoríme pracovné prostredie, ktoré bude zobrazovať prijaté správy jednotne.

## **Kľúčové slova**

Sjednocena komunikace; Zjednotená komunikácia; Unified Communications; UC; VOIP; XMPP; Asterisk; Voice glue; Voice xml; POP3; IMAP; SMTP; IM; Email; TTS; text-to-speech; PHP;

## **Abstract**

This diploma thesis deals with problematics with united communication. There are described different problems and consequences which rise with using several types of communication channels. We say something about the most used communication protocols. We create platform designs, which will be uniting required protocols to uniform system. In this scheme we look closer to problems which rise here and we will suggest solutions which eliminate this problems. We defined requirement that the system, which will solve the UC, must fulfill. With help with this create scheme we will implement mechanisms for system core and individual modules. We will create web user interface and integrate services in it, such as transformation text to voice. In this interface will be create work environment which will displayed uniting received message.

## **Key words**

Unified Communications; UC; VoIP; XMPP; Asterisk; Voice glue; Voice xml; POP3; IMAP; SMTP; IM; Email; TTS; text-to-speech; PHP;

## Zoznam použitých skratiek

Skratka	Význam
<b>XMPP</b>	Extensible Messaging and Presence Protocol
<b>VoIP</b>	Voice over Internet Protocol
<b>UC</b>	Unified Communications
<b>POP3</b>	Post Office Protocol v3
<b>IMAP</b>	Internet Message Access Protocol
<b>TTS</b>	Text-To-Speech
<b>SMTP</b>	Simple Mail Transfer Protocol
<b>IM</b>	Instant Messaging
<b>HTTP</b>	Hypertext Transfer Protocol
<b>PHP</b>	Hypertext Preprocessor
<b>IP</b>	Internet Protocol
<b>AJAX</b>	Asynchronous JavaScript and XML
<b>TCP</b>	Transmission Control Protocol
<b>UDP</b>	User Datagram Protocol
<b>AGI</b>	Asterisk Gateway Interface
<b>API</b>	Application Programming Interface
<b>URL</b>	Uniform Resource Locator
<b>XML</b>	Extensible Markup Language
<b>XHTML</b>	Extensible Hypertext Markup Language

# Obsah

1	Úvod do problematiky zjednotenej komunikácie UC .....	- 2 -
1.1	Problematika UC .....	- 2 -
1.2	Unified Communications .....	- 3 -
1.3	Zariadenia používané v UC .....	- 3 -
2	Komponenty UC pre služby .....	- 5 -
2.1	Voip .....	- 5 -
2.1.1	Asterisk .....	- 5 -
2.1.2	PHP AGI .....	- 5 -
2.1.3	Asterisk Gateway Interface (AGI) .....	- 5 -
2.2	Text-to-speech .....	- 5 -
2.2.1	VoiceXML .....	- 6 -
2.2.2	VoiceGlue .....	- 6 -
2.2.3	Festival .....	- 6 -
2.2.4	Google TTS .....	- 6 -
2.3	E-mail .....	- 7 -
2.3.1	POP .....	- 7 -
2.3.2	IMAP .....	- 7 -
2.3.3	SMTP .....	- 7 -
2.4	Textové správy .....	- 7 -
2.5	Instant Messaging .....	- 8 -
2.5.1	XMPP .....	- 8 -
3	Návrh a realizácia platformy .....	- 9 -
3.1	Požadované vlastnosti platformy .....	- 9 -
3.1.1	Mobilita a flexibilita platformy .....	- 9 -
3.1.2	Jazyková podpora .....	- 9 -
3.1.3	Modularita systému .....	- 9 -
3.1.4	Model AAA .....	- 10 -
3.1.5	Komunikačné zariadenia .....	- 10 -
3.1.6	Spracovanie hlasu TTS .....	- 10 -



3.1.7	Užívateľské rozhranie.....	- 11 -
3.1.8	Manažment zdrojov.....	- 12 -
3.1.9	Podporované protokoly .....	- 12 -
3.1.10	Jednotná forma správ.....	- 12 -
3.1.11	Bezpečnosť komunikácie a firemných sprav.....	- 12 -
3.1.12	Integrovanie telefónie.....	- 12 -
3.2	Návrh koncepcie.....	- 13 -
3.2.1	Jadro platformy.....	- 14 -
3.2.2	IM a poštový modul .....	- 16 -
3.2.3	TTS a hlasový modul .....	- 17 -
3.2.4	Integrácia telefónnych služieb .....	- 19 -
3.2.5	Užívateľské rozhranie.....	- 21 -
3.2.6	Modul smerovania komunikácie .....	- 23 -
3.3	Vytvorenie rôznych častí platformy .....	- 25 -
3.3.1	Jadro systému .....	- 25 -
3.3.2	Webové grafické rozhranie.....	- 26 -
3.3.3	Prezencia .....	- 31 -
3.3.4	Text To Speech.....	- 35 -
4	Zhodnotenie dosiahnutých výsledkov .....	- 38 -
4.1	Ďalšie časti systému .....	- 38 -
4.2	Stromová architektúra priečinkov a pod priečinkov .....	- 39 -
	Záver .....	- 41 -
	Použitá literatúra .....	I
	Zoznam príloh .....	II

## Úvod

Medzi najvýraznejšie trendy v podnikovej komunikácii jednoznačne patrí zdieľanie a plánovanie prenesené do mobilného prostredia s možnosťou pristupovať ku kľúčovým informáciám kedykoľvek a kdekoľvek. Najvyšší efekt na skvalitnenie komunikácie v rámci podniku má dôsledná práca s ľuďmi a každodenná starostlivosť o otvorenosť, zrozumiteľnosť a zmysluplnosť dorozumievania sa medzi kolegami a klientmi. Moderné technológie sú tu vítaným nástrojom a pomocníkom - najmä technológie zaisťujúce komunikáciu ľudí bez ohľadu na to, kde sa práve nachádza a aký nástroj komunikácie majú k dispozícii. Všetko takto popísané nazývame zjednotenou komunikáciou angl. (Unified Communications - UC).

Unified Communications je rozvíjajúci sa prístup ku komunikácii, ktorý rieši nespočetné množstvo otázok v modernom mobilnom pracovnom prostredí. Táto kapitola popisuje existujúce komunikačné problémy. Tiež ukazuje, ako Unified Communications stratégia začína transformovať tieto roztrieštene technológie do uceleného riešenia.

V tejto diplomovej práci sa budeme zaoberať problematikou UC. Kde si povieme o problémoch ktoré vznikajú pri používaní rôznych druhov komunikačných kanálov. Pokúsime sa vytvoriť technologické riešenie týchto problémov. Definujeme ktoré komponenty a protokoly budeme integrovať do systému, ktorý zjednocuje komunikáciu. Vytvoríme teoretický návrh kde platformu rozdelíme do jednotlivých funkčných blokov a popíšeme spôsob ako tieto bloky medzi sebou komunikujú. Bližšie sa pozrieme aké funkcie a mechanizmy obsahujú tieto bloky. V praktickej časti budeme implementovať teoretický návrh do funkčných časti systému. Integrujeme tu poštové a hlasové služby ktoré budeme používať prostredníctvom webového rozhrania. Webové rozhranie nám bude slúžiť ako pracovne prostredie pre zjednotene zobrazovanie prijatých správ. Toto prostredie by malo spĺňať nami dane požiadavky ktoré vychádzajú s najmodernejších trendov používaných pri implementácii užívateľského rozhrania. Cieľom tejto práce by mal byť z časti funkčný systém ktorý bude obsahovať pracovne prostredie ktoré budeme mať implementované mechanizmy autentifikácie a autorizácie. Pracovne prostredie bude ďalej obsahovať zjednotene zobrazenie prijatých správ kde budeme mať možnosť preložiť text správy na hlas a pomocou integrovaného prehrávača si budeme moc tuto zvukovú stopu vypočuť. Vytvorené funkčne časti platformy spolu s databázou poskytneme ako prílohu k tejto práci.

# 1 Úvod do problematiky zjednotenej komunikácie UC.

## 1.1 Problematika UC

Dnešná doba vyvoláva nový spôsob pohľadu na staré problémy, a to je to, s čím sa firmy budú musieť vysporiadať v prípade internej a externej komunikácie. Spoločnosti sa potrebujú posunúť v pohľade zameraním na riešenie komunikácie so špeciálnymi funkciami pre konečného užívateľa zariadení na aplikácie pre zjednodušenie komunikácie. A zabezpečiť, aby ľudia mohli začať, prijímať a spravovať komunikáciu, kedy, kde, ako a s kýmkoľvek sa im zapáči, bez ohľadu na to aby sa museli učiť používať nové komplikované technológie. Vďaka integrácii komunikácie cez široké spektrum režimov, aplikácií a zariadení sa všetky systémy stávajú viac užívateľsky prívetivé. V informačnej dobe sa kladie veľký dôraz na získavanie a výmenu informácií. Rýchly prístup k dôležitým informáciám poskytuje kľúčovú výhodu v konkurenčnom boji. Preto do komunikácie medzi zamestnancami vo firemnom prostredí alebo medzi zamestnancami a klientmi, spoločnosti investujú veľké množstvo kapitálu, aby si túto konkurenčnú výhodu udržali. Na trhu sa nachádza veľké množstvo platených a voľne šíriteľných komunikačných technológií, preto vzniká problém s výberom vhodnej komunikačnej technológie, na ktorú sa treba primárne zamerať. Firmy sú nútené investovať svoje zdroje do rôznych technológií, ktoré nie sú medzi sebou kompatibilné. Túto nekompatibilitu sa snažia vykompenzovať vyvíjaním vlastných riešení, ktoré integrujú tieto technológie. Integrácia rôznych komunikačných technológií sa označuje ako zjednotená komunikácia.

- Náklady na správu a údržbu rôznorodých sietí, aplikácií a zariadení môžu byť vysoké, nehovoriac o časovej náročnosti. Všetky komunikačné zariadenie zamestnancov treba aktualizovať a udržiavať ich zabezpečené, to vytvára veľké požiadavky na kvalitu IT pracovníkov.
- Súkromná komunikácia pre firemne účely. Keď pracovníci komunikujú so zákazníkmi cez prostriedky, ktoré im ich spoločnosť nezabezpečila (mobilný telefón, spotrebiteľ IM, Internet), bude táto spoločnosť riskovať, že komunikácia medzi zákazníkmi, partneri, dodávateľmi a zamestnanci, ktoré sa uskutočňujú cez populárne "verejné" kanály, nebude pod kontrolou.
- Nebezpečenstvo úniku informácií. Väčšina z týchto "verejných" kanálov má menšiu ochranu ako spoločnosti požadujú: služby IM sú nešifrované, osobné e-mail sú uložené na serveroch s viacerými vlastníkmi. Väčšina týchto informácií podlieha ochrane údajov, ich predpisom a zásadám uchovávania.
- Nedokumentovaná komunikácia. Podniky sú pod zvyšujúcimi sa požiadavkami na dokumentáciu ich vnútornej komunikácie rovnako aj externej komunikácie so zákazníkmi aj dodávateľmi. Ak táto komunikácia prebieha cez verejné kanály, možnosti dokumentácie tejto komunikácie sú veľmi obmedzené.

## 1.2 Unified Communications

Veľký nárast dopytu je po zjednotení komunikačných riešení vo firmách tak, aby sa zvýšila efektívnosť práce bez nutnosti investovať do jednotlivých proprietárnych riešení. Veľkú úlohu v tom zohráva aj súčasná finančná kríza, ktorá núti podniky hľadať riešenie s vysokou pridanou hodnotou za prijateľnú cenu. Jednotná komunikačná infraštruktúra umožňuje preniesť všetky druhy komunikácie ako sú dátové, hlasové služby a video služby, do spoločného komunikačného kanálu. Zlúčená sieť dovoľuje pracovať spôsobom, ktorý najlepšie vyhovuje potrebám, a to pomocou drôtových aj bezdrôtových telefónov, e-mailu, faxov, videokonferencií a kontaktných centier.

Hlavnou myšlienkou UC je spájanie komunikačných kanálov do jedného uzla a v prípade potreby umožňuje redistribúciu medzi jednotlivými kanálmi, respektíve medzi jednotlivými protokolmi. Jednou z hlavných vlastností systému zjednotenej komunikácie je jednotný formát všetkých prijatých a odoslaných správ. Aby mohla byť splnená táto vlastnosť, musí systém podporovať mechanizmy prevodu správ z jednotlivých protokolov na jednotný formát. Tento formát má niekoľko výhod, jednou z nich je použitie jednotného mechanizmu pre ďalšie spracovanie správ tohoto typu, to znamená že, nie sú potrebné rôzne metódy spracovania pre jednotlivé podporované protokoly, prenášajúce tieto správy. Ďalšou nespornou výhodou je zber správ z jednotlivých protokolov jedným systémom a nasledovné ukladanie na jedno miesto, to znamená, že užívateľ nepotrebuje používať klientske aplikácie pre rôzne formy komunikácie ale prístupuje k týmto správam jednotnou formou podľa toho, ako si to sám primárne určil. Cieľom tejto práce je navrhnutie konceptu, jednotlivých metód a mechanizmov potrebných pre vytvorenie platformy, ktorá získava prijaté správy na jednotlivých komunikačných účtoch a prevádza ich do jednotného formátu. Kľúčovým faktorom je zjednotený a jednoduchý spôsob obsluhy všetkých týchto komunikačných kanálov a firemných služieb. Tak, aby komunikácia prácu uľahčovala.

## 1.3 Zariadenia používané v UC

Pracovníci dnes majú mnoho prostriedkov pre komunikáciu. Spoločnosti im poskytujú: kancelársky telefón, hlasovú poštu, fax, e-mail, a niekedy aj instant messaging, audio konferencie, a teraz Web a videokonferencie. Jednotlivo, každý spôsob komunikácie funguje dobre pre určité potreby, ale celkovo sa ukázali ako nedostatočné pre mnohé účely. E-mail je rozhodne rýchlejší ako staromódny list, takže ľudia majú zvyčajne vysokú tendenciu reagovať na e-mail, ale často to nedosahuje. Ľudia sú zaplavení príliš veľkým množstvom e-mailových správ. Ľudia taktiež nie sú nalepení na kancelárskej stoličke, sú zaneprázdnení volaním po celý deň, takže telefónne volanie sú optimálne pre rýchlu odpoveď.

- Stacionárne telefónny / pevná linka: Dnes už nemôžeme pod týmto pojmom chápať len analógové a digitálne telefóny kde poskytovateľ služieb je národný telekomunikačný operátor, do tejto kategórie patria aj telefóny na báze voip. Voip telefónnu ústredňu môžeme prevádzkovať bez veľkých prevádzkových nákladov aj lokálne alebo pomocou internetového poskytovateľa.

- Mobilné telefóny: Počet mobilných zariadení celosvetovo v roku 2013 presiahol hranicu 6.8 miliardy preto, je ľahké predpokladať, že väčšina administratívnych pracovníkov používa mobilný telefón. U poskytovateľa má každý užívateľ možnosť využívať služby hlasovej pošty.
- Smartphony a PDA: Možnosti mobilných telefónov a PDA sa zblížila a vytvorila novú generáciu inteligentných telefónov, ktoré sú schopné prístupu na Internet, odosielanie a prijímanie e-mailov, vedenie kalendára a zoznamy kontaktov a ich skladovanie. Pracovníci si často kupujú jeden z týchto zariadení pre osobné a obchodné využitie.

## 2 Komponenty UC pre služby

Pod pojmom komponenty zjednotenej komunikácie budeme rozumieť všetky použité aplikácie, programy, protokoly, moduly, programovacie jazyky potrebné na prepojenie jednotlivých blokov. Ďalej to môžu byť služby poskytované určitými poskytovateľmi a rôzne technológie a riešenia, ktoré napomáhajú k skvalitneniu a zefektívneniu poskytovaných služieb v UC.

### 2.1 Voip

Riešenie voip je neoddeliteľnou súčasťou zjednotenej komunikácie, ktorá pomáha integrovať hlas, video, dáta a mobilné aplikácie na pevných alebo mobilných sieťach, a to z ktoréhokoľvek média, zariadenia alebo operačného systému. IP telefónia ponúka spôsob, ako rozšíriť konzistentné komunikačné služby v rámci IKT pre všetkých zamestnancov. Prenáša komunikáciu cez sieť s použitím otvorených štandardov založených na internetovom protokole. Táto technológia zaisťuje vedľa vlastných výhod konvergovanej siete väčšiu bezpečnosť, odolnosť a škálovateľnosť riešenia.[3]

#### 2.1.1 Asterisk

Asterisk umožňuje ľahko vytvárať komunikačné aplikácie. Začalo to ako jadro systému pobočkovej ústredne (PBX) a dnes na tomto jadre stavajú státisíce telefónnych systémov po celom svete. Ale telefónne systémy sú len začiatok. Dnes má Asterisk silu radu komunikačných aplikácií, vrátane automatického distribútora hovorov, konferenčného mosta, hlasovej schránky, zjednotene zasielanie správ a ďalšie. [1][3]

#### 2.1.2 PHP AGI

Je to súbor metód, mechanizmov napísaných v programovacom jazyku PHP určených pre Asterisk Gateway Interface (AGI). Balíček je k dispozícii na použitie a distribúciu pod podmienkami licencie GNU Public License. PHPAGI bol pôvodne napísaný Matthew Ashamom a bol opravený, vylepšený a tvarovaný tímom vývojárov.[9]

#### 2.1.3 Asterisk Gateway Interface (AGI)

Asterisk Gateway Interface je rozhranie prídavných funkcií pre Asterisk ktoré môžeme ovládať pomocou mnohých rôznych programovacích jazykov. Perl, PHP, C, Pascal, Bourne Shell.

### 2.2 Text-to-speech

Text-to-speech, je konverzia napísaného slova do hlasového výstupu pomocou syntézy reči, umelý spôsob, ako umožniť prístup k verbálnej komunikácii v alternatívnej podobe. Text-to-speech technológie využívajú softvér a hardvér, ktorý zaisťuje hlasový výstup z textového vstupu. Táto technológia môže vytvoriť slová z uložených foném, jednotlivých prvkov reči a tiež

celých slov a fráz pre dosiahnutie čo najprirodzenejšieho hlasu. Text-to-speech technológie majú širokú škálu aplikácií.

TTS systém sa skladá z dvoch častí: front-end a back-end. Front-end má dve hlavné úlohy. Po prvé, prevádza surový text obsahujúci symboly ako čísla a skratky do ekvivalentu písomnej podoby. Tento proces sa často nazýva normalizácia textu, pre-spracovanie. Front-end priradí fonetické prepisy ku každému slovu, predelí a označí text do prozodických jednotiek, ako fráza, klauzula a vety. Proces určenia fonetických prepisov na slová sa nazýva text-to-fonéma či graf-to-fonéma konverzia. Fonetická transkripcia a prozódia spoločne tvoria symbolickú lingvistickú reprezentáciu, ktorá je výstup z front-end. Back-end, často sa nazýva ako syntetizátor, premieňa symbolickú lingvistickú reprezentáciu do zvuku. V systémoch, táto časť obsahuje výpočet cieľovej prozódie, ktorá je potom zohľadnená na výstupnej reči.[8]

### 2.2.1 VoiceXML

Je značkovací jazyk určený na popis interaktívnych dialógov medzi človekom a počítačom, je vyvinutý na báze XML, ktorý vyvíja konzorcium W3C. VoiceXML je určený na vytváranie scenárov pre automatizovaný dialóg medzi človekom a počítačom, je navrhnutý pre hlasové dialógy, rozpoznávaním hovoreného vstupu od užívateľa a syntetizáciou výstupu. Umožňuje tiež rozpoznávať DTMF voľby a vytvárať telefónne spojenie. Obsahuje výstup vo forme zvukových súborov, ktoré môžu nahradiť syntetizovaný výstup, ten je tvorený pomocou modulov Text-To-Speech.[8]

### 2.2.2 VoiceGlue

VoiceGlue je rozšírenie pre Asterisk ústredňu, ktorý umožňuje spracovávanie aplikácií VoiceXML. Je založený na open source riešeniach a je vydávaný pod licenciou GNU GPL, využíva OpenVXI interpreter. V distribúcii je zahrnutý syntetizátor Flite, VXML 2.0 interpret s niektorými funkciami VXML 2.1, SRGS DTMF gramatiky a ďalšie. Projekt VoiceGlue vznikol za účelom získať lacnejšie riešenie VoiceXML aplikácie. Hlavným autorom a vývojárom je Doug Campbell zo spoločnosti Ampersound.[8]

### 2.2.3 Festival

Festival systém syntézy reči, je všeobecne viac jazyčný systém syntézy zvuku pôvodne vyvinutý Alanom W. Black na Centrum pre výskum reči (CSTR) na univerzite v Edinburghu. Je distribuovaný pod BSD licenciou. Ponúka kompletný prevod textu na reč systému s rôznymi API, rovnako ako prostredie pre vývoj a výskum techník syntézy reči. Je napísaný v C++. Festival je určený pre podporu viacerých jazykov a prichádza s podporou angličtiny (britská a americká výslovnosť), Waleštiny a španielčiny. Hlasové balíky existujú pre niekoľko ďalších jazykov, ako češtinu, fínčinu, hindčinu, taliančinu, poľštinu a ruštinu.[8]

### 2.2.4 Google TTS

Hlasový modul, ktorý používa spoločnosť Google u svojich produktov. Spracovanie reči je na vysokej úrovni a stále stúpa, podporuje rôzne druhy jazykov. Prístupovať k tomuto TTS

modulu môžeme pomocou rodiny http protokolov. Do URL reťazca vložíme požadovaný vstupný text, jazyk a ďalšie parametre. Odošleme na spracovanie a hlasový modul nám poskytne zvukovú stopu vytvoreného hlasu.

## 2.3 E-mail

Väčšina pracovníkov má aj osobné e-mailové kontá, od poskytovateľov ako AOL, Yahoo, Gmail, Hotmail, MSN a mnoho ďalších. Hoci mnoho pracovníkov, má oddelené e-maily na osobné a podnikateľské účely, niekedy sa môžu zmeniť z osobných e-mailov na podnikateľské účely, ak ich spoločnosť nemá k dispozícii vlastne e-mailové schránky.

### 2.3.1 POP

Post Office Protocol (POP) je internetový protokol na aplikačnej vrstve, ktorý sa využíva na prijímanie elektronickej pošty zo vzdialeného servera prostredníctvom TCP/IP spojenia. Poštový protokol je séria pravidiel o tom, ako sa má riadiť prenos elektronickej pošty medzi dvomi bodmi v sieti. POP3 je následníkom protokolov POP1 a POP2 (označenie POP už dnes takmer výhradne znamená POP3).[4]

### 2.3.2 IMAP

Internet Message Access Protocol (IMAP) je internetový protokol umožňujúci prístup k e-mailovým schránkam. V súčasnosti sa používa verzia IMAP4 (IMAP version 4 revision 1 – IMAP4rev1), definovaná v RFC 3501.

Na rozdiel od protokolu POP3 je optimalizovaný pre prácu v dlhodobom pripojenom režime, keď správy zostávajú uložené na serveri a priebežne sa sťahujú, keď je to potrebné. Rozdiely zahŕňujú podporu pre prácu viacerých pripojených klientov zároveň, uchovávanie stavov správ na serveri, podporu niekoľkých zložiek a prehľadávanie správ na strane servera.[5]

### 2.3.3 SMTP

Jednoduchý protokol na prenos pošty (angl. simple mail transfer protocol), skr. SMTP, je jednoduchý protokol umožňujúci prenos e-mailov medzi stanicami. Protokol zaisťuje doručenie pošty pomocou priameho spojenia medzi adresátom a odosielateľom; správa je doručená do tzv. poštovej schránky adresáta, ku ktorej môže užívateľ ďalej pristupovať, napr. pomocou protokolov POP3 a IMAP. Ide o jednu z najstarších aplikácií, pôvodná norma RFC 821 bola vydaná v roku 1982. SMTP funguje nad protokolom TCP a na strane serveru štandardne využíva port 25.[6]

## 2.4 Textové správy

Používatelia mobilných telefónov môžu posilať textové správy sebe navzájom. Hlavní poskytovateľom je mobilný operátor alebo internetová brána textových správ.



## 2.5 Instant Messaging

Pre tých, ktorí si myslia že e-mail nie je dosť rýchly, a ak podnik neprevádzkuje IM služby od vlastných dodávateľov softvéru (ako je napríklad Microsoft, Jabber), tak sú zamestnanci nútení si zriadiť vlastné služby (ako je MSN, Yahoo, Google a Skype). Niektoré z týchto riešení IM teraz dovoľujú na internete založené telefónne hovory a dokonca aj to aby užívatelia chatu mohli telefonovať na pevné a mobilné linky.

### 2.5.1 XMPP

Extensible Messaging and Presence Protocol (XMPP) (predtým známy ako Jabber) je protokol používaný na sieťovú komunikáciu. XMPP je súbor protokolov a technológií založených na XML, ktorá umožňuje akýmkoľvek dvom entitám na internete, vzájomne si vymieňať textové správy, informácie o prítomnosti a ďalšie štruktúrované informácie v (takmer) reálnom čase.

Základné protokoly (XMPP RFC):

- RFC 3920: Rozšíriteľný protokol posielania správ a stavu prítomnosti (Extensible Messaging and Presence Protocol) (XMPP): Jadro technológie XML streaming, ktorý je základom XMPP aplikácií a obsahuje pokročilú bezpečnosť a viacjazyčnú podporu.
- RFC 3921: Rozšíriteľný protokol posielania správ a stavu prítomnosti (Extensible Messaging and Presence Protocol) (XMPP): Základné rozšírenia pre okamžité správy a stav prítomnosti, vrátane zoznamu kontaktov, posielanie/prijímanie stavu prítomnosti a umiestnenie na bielu/čiernu listinu.
- RFC 3922: Mapovanie XMPP do CPIM - Mapovanie XMPP do abstraktnej syntaxe IETF na okamžité správy a stav prítomnosti.
- RFC 3923: Medzi koncové podpisovanie a šifrovanie objektov pre XMPP - Rozšírenie na medzi koncovú vzájomne funkčnú bezpečnosť.[7]

## 3 Návrh a realizácia platformy

Pri návrhu sa na UC pozrieme z technického hľadiska, tento systém, aplikáciu budeme nazývať platforma pretože má funkciu spojiva medzi jednotlivými systémami, aplikáciami, protokolmi. Tato platforma musí mať takú úroveň sofistikovanosti aby dokázala zabezpečiť zber správ a informácií z jednotlivých komunikačných kanálov, správne smerovanie medzi jednotlivými kanálmi a komplexný manažment. Všetky tieto činnosti musia byť zabezpečené v čase, ktorý je určený užívateľmi, preferenciami alebo podľa rozhodovacích mechanizmov jednotných globálnych alebo lokálnych politik. Kvôli komplexnosti platformy musíme pre lepší návrh rozdeliť do jednotlivých blokov, mechanizmov, podsystémov.

### 3.1 Požadované vlastnosti platformy

#### 3.1.1 Mobilita a flexibilita platformy

Vysoká mobilita a flexibilita je primárna vlastnosť každého UC systému. Pod pojmom mobilita môžeme chápať možnosť mať prístup ku svojej komunikácii kedykoľvek a kdekoľvek, pri použití rôznych komunikačných zariadení, bez ohľadu na typ zariadenia tak, aby mal používateľ prístup ku čo najväčšiemu počtu služieb poskytovaných UC systémom. Väčšinu úkonov potrebných pre manažment komunikačných kanálov preberá na seba systém UC, to znamená, že koncového užívateľa odbremení od základných úkonov potrebných pre používanie rôznych komunikačných kanálov. Táto vlastnosť redukuje čas a prostriedky potrebné na fungovanie rôznych komunikačných kanálov.

#### 3.1.2 Jazyková podpora

Medzi základné vlastnosti každého moderného komunikačného systému je podpora rôznych jazykov. V našom prípade to musíme chápať ako podporu jazykov hovoreného slova TTS a podporu jazykov užívateľského rozhrania systému. Treba zabezpečiť aby výber jazyka bol v kompetenciách každého užívateľa zvlášť, aby inštalácia nového jazykového balíčka bola užívateľsky nenáročná a nepotrebovala by veľký zásah do funkčnosti systému. V dôsledku týchto problémov je najlepšia taká voľba kde sa o podporu jazykov stará oddelený jazykový modul. Vstupné jazykové listiny (baličky) musia mať jednotnú formu pre efektívnejšie a jednoduchšie uprávi.

#### 3.1.3 Modularita systému

Aby sme zabezpečili vysokú flexibilitu a dynamiku systému, rýchly vývoj a implementáciu nových modulov a služieb, systém musí byť na báze modulárnej architektúry kde väčšie skupiny mechanizmov, služieb a podsystémov sú rozdelené do určitých modulov podľa rôznych kritérií tak, aby spĺňali základné poslanie tejto architektúry. Rozdeľovanie do jednotlivých celkov sa musí uskutočňovať na rôznych úrovniach, a to sú napríklad:

- oddelenie obsahu od grafického rozhrania, podpora viacerých jazykov
- oddelenie grafického rozhrania od systémových funkcií modulov

- oddelenie základných funkcií od pokročilých

### 3.1.4 Model AAA

Služby, ktoré poskytuje UC systém nie sú určené pre akéhokoľvek návštevníka. Systém musí zabezpečiť aby služby poskytované systémom mohli využívať iba oprávnené osoby. Mechanizmy zaručujúce poskytovanie služieb oprávneným osobám sa súhrnne nazývajú AAA protokol. AAA v skratke znamená (Authentication, Authorization and Accounting), v preklade to znamená autentizácia, autorizácia a účtovanie. Autentizácia je súbor mechanizmov ktoré zabezpečujú overenie identity užívateľa, či je to osoba, za ktorú sa vydáva. Autorizácia je mechanizmus pridelenie užívateľských oprávnení, ktoré mu boli pridelené oprávnenou osobou za určitých podmienok. Tento mechanizmus musí zaručiť to že každý užívateľ môže konať v rámci svojich kompetencií. Vyúčtovanie poskytnutých služieb musí byť taktiež neoddeliteľnou súčasťou takéhoto systému, aby sme mohli využívať takýto nástroj musíme ukladať záznamy o poskytnutých službách preto, aby sme dokázali určiť o aký druh služby sa jedná a v akej miere bola využitá.

### 3.1.5 Komunikačné zariadenia

Doba, kedy sme mohli internet používať iba pomocou počítača je už dávno preč, dnes existuje široká škála rôznych zariadení, ktoré majú prístup na internet, sú to napríklad inteligentné telefóny, tablety, internet v aute, v budúcnosti to budú inteligentne hodinky a okuliare. Musíme si uvedomiť, že každé zariadenie má svoje špecifiká, ako sú napríklad výpočtový výkon hardwaru, veľkosť zobrazovacej plochy, hardwarové alebo softwarové tlačidlá, typ operačného systému a vlastnosti prenosového média. UC systém musí podporovať širokú paletu takýchto zariadení pretože to je jedna zo základných vlastností UC. Pri používaní rôznych druhov zariadení nastáva problém, že každé zariadenie potrebuje individuálny prístup k systému kvôli rôznym požiadavkám na grafický alebo komunikačný výstup systému, to znamená, že systém musí obsahovať mechanizmus, ktorý spoľahlivo zabezpečí detekciu typu zariadenia a upraví výstupný formát pre každé zariadenie zvlášť. Pre lepšie poskytovanie služieb musí systém rozoznať s ktorým zariadením užívateľ momentálne komunikuje, tento mechanizmus zabezpečí to, že napríklad telefónny hovor bude presmerovaný na používané zariadenie automaticky, ak si to užívateľ praje. Táto schopnosť je ďalšou požadovanou vlastnosťou systému, ktorá nám zabezpečí efektívne smerovanie nielen medzi komunikačnými kanálmi ale aj medzi jednotlivými zariadeniami.

### 3.1.6 Spracovanie hlasu TTS

Grafické rozhranie už nie je jediným komunikačným médiom medzi vyspelým počítačovým systémom a človekom, dnes tu máme aj hlbšiu integráciu hlasových služieb. Dynamická doba napomáha intenzívnejšiemu využitiu rýchlej hlasovej služby, pred textovou formou komunikácie. Hlavnou výhodou takejto služby je rýchlosť a objem výmeny informácií, to taktiež znamená že šetrí čas. Hlasovú komunikáciu medzi vyspelým zariadením a človekom môžeme rozdeliť na dva smery, jeden smer je hlas ktorý je na vstupe hlasového rozhrania, toto

rozhranie interpretuje tento hlas do formy s ktorou toto zariadenie môže pracovať. Druhý smer je hlas na výstupe hlasového rozhrania, ktoré transformuje grafický alebo textový formát na hlas, tento prevod sa nazýva TTS (text to speech). Oba smery sa momentálne nachádzajú na začiatku vývoja, na trhu sa nachádzajú rôzne technológie na prevod hlasu a textu medzi sebou. Tieto formy prevodu môžeme považovať za primitívne pretože tieto systémy nedokážu reprodukovat' hlas takým spôsobom ako to robí človek, pracujú so vzorkami hlasov, ktoré sú uložené v databáze, ktorú musel vytvoriť človek, tieto vzorky systém skladá podľa určitých algoritmov do viet, súvetí, textu. Ďalej, do týchto slov nedokážu vložiť emócie a pocity. Z hlasom na vstupe je to trochu horšie, je tu veľa faktorov ktoré treba odstrániť aby sa minimalizovala chybovosť systému. Momentálne dostupné technológie pracujú tak, že nedokážu rozumieť hlasovému prejavu ako celku ale rozdelia zaregistrovaný hlas na určitý počet hlasových vzoriek a tie potom porovnávajú s vnútornou databázou podľa rôznych kritérií. Musíme si uvedomiť to, že každý ľudský hlas je iný po rôznych stránkach preto je veľmi náročné vytvoriť takú databázu vzoriek, ktorá bude zohľadňovať všetky potrebné kritériá. Takéto systémy nám momentálne rozumejú iba vtedy keď použijeme kľúčové slova, ktoré podľa určitých postupov skladáme tak, aby nám vznikla určitá forma hlasovej správy, ktorú tento systém dokáže dekódovať. Hlasové systémy sú veľmi komplexné, preto treba vynaložiť veľa zdrojov na ich vývoj. Takýto vývoj si môžu zatiaľ dovoliť iba vyspelé technologické spoločnosti ako je Google, Apple, Microsoft a iný, taktiež tu máme aj rôzne menšie open-source projekty ale tie zatiaľ neposkytujú takú zrozumiteľnosť a kvalitu ako tieto spoločnosti.

### 3.1.7 Užívateľské rozhranie

Je práve to rozhranie, ktoré reálne spája človeka a počítačový systém, pomocou preddefinovaných postupov si navzájom vymieňame informácie a požiadavky. Užívateľské rozhrania sa môžu deliť podľa viacerých kritérií, nás hlavne zaujíma delenie podľa formy komunikácie, kde máme dve možnosti, grafická a hlasová forma, ďalej tu máme delenie podľa typu koncového prístupového zariadenia, koncové zariadenie môže pristupovať pomocou webového rozhrania alebo pomocou aplikácie určenej pre konkrétne zariadenie, ktorá lepšie využíva natívne funkcie zariadenia. Web je skôr univerzálne rozhranie, pretože je to jedna z najpoužívanejších technológií. Grafické rozhranie musí byť optimalizované pre rôzne veľkosti obrazoviek a taktiež pre rôzne typy internetových prehliadačov. Pre lepšiu, jednoduchšiu a rýchlejšiu prácu s takýmto rozhraním musí tento systém podporovať najnovšie trendy v tejto oblasti. Takýto systém sa väčšinou skladá z viacerých častí ktoré sú vytvorené pomocou rôznych technológií, napríklad pre lepšiu dynamiku rozhrania sa používajú technológie ako su AJAX, jQuery a cURL. Pri návrhu takéhoto rozhrania musíme mať na vedomí to, že toto rozhranie musí byť oddelené od mechanizmov systému, ďalej tu máme jazykovú podporu, ktorá musí byť taktiež oddelená od systému a tiež nasadenie nového jazyka musí byť jednoduché a nemôže zasahovať do funkčnosti systému.

### 3.1.8 Manažment zdrojov

Jednou z hlavných požiadaviek na UC je zdokumentovateľnosť vnútornej aj vonkajšej komunikácie. Systém môže zaznamenávať aj iné informácie nielen komunikáciu, môžu to byť informácie o časovom rozvrhu zamestnancov alebo aj aké služby boli poskytnuté jednotlivým užívateľom. Takéto množstvo dát nám môže poskytovať viac informácii ako je obsiahnuté v týchto dátach, pomocou implementácie mechanizmov môžeme tieto dáta štatisticky analyzovať a získavať informácie o chovaní zamestnancov, o stave poskytovaných služieb, o toku informácii v rámci systému a tiež môžeme predvídať ďalšie skutočnosti. Pokročilé analyzátory dát nám poskytujú lepšie spätnú väzbu o fungovaní systému, o správaní sa zamestnancov a o kvalite poskytovaných služieb.

### 3.1.9 Podporované protokoly

Každá komunikačná technológia ma svoj vlastný protokol pomocou ktorého prenáša potrebné informácie. UC platforma musí podporovať väčšinu bežne používaných komunikačných protokolov. Každý protokol musí mať vlastnú sadu mechanizmov na manažment, prijímanie a odosielanie správ pomocou tohoto protokolu. Systém musí tiež obsahovať mechanizmus na preposielanie správ medzi jednotlivými protokolmi. Tento mechanizmus konvertuje formu prijatej správy do takej formy, ktorá spĺňa konkrétne špecifikácie jednotného formátu, taktiež to musí dokázať konvertovať spätne. Najdostupnejšie protokoly pre Instant Messaging a email sú: XMPP pre IM, POP3, IMAP pre prijatie a SMTP pre odosielanie pošty.

### 3.1.10 Jednotná forma správ

Základom každého UC systému je ukladanie správ do jednotného formátu. Všetky formy komunikácie sa ukladajú do spoločnej tabuľky v jednej databáze. Všetky záznamy musia uchovávať nielen obsah správy, ale aj protokol pomocou ktorého bola správa doručená. Jednotná forma nám zaručí jednotný prístup k takejto forme komunikácie a jednoduchší prevod textu na hlas.

### 3.1.11 Bezpečnosť komunikácie a firemných správ

Väčšina firiem nemá dohľad nad vnútornou a vonkajšou komunikáciou svojich zamestnancov. Je to spôsobené používaním rôznej technológie, ktorá neobsahuje kontrolné mechanizmy. Spoločnosti, ktoré nemajú možnosť kontroly komunikačných kanálov využívaných vlastnými zamestnancami, môžu vytvárať bezpečnostné rizika, ktoré spôsobujú vedomí alebo nevedomý unik firemných informácií a správ. Dokumentácia všetkej komunikácie nám môže poskytovať cennú spätnú väzbu o tom, ako efektívne sa využíva firemná komunikácia a taktiež, nám povie niečo o tom, ako sa správajú zamestnanci voči partnerom a zákazníkom. Takýto zber údajov a informácií podlieha miestnym zákonom o spracovaní osobných údajov.

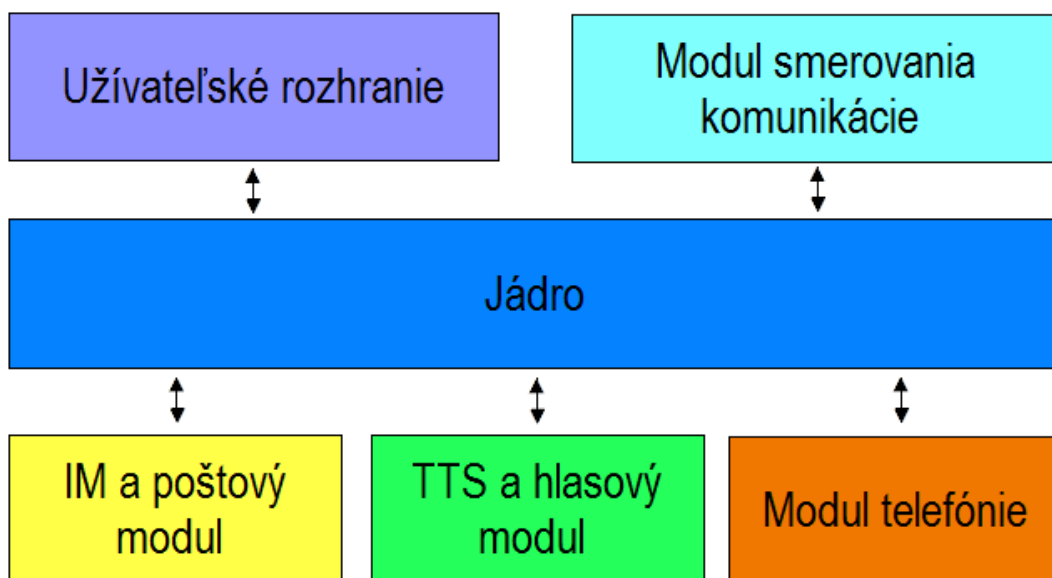
### 3.1.12 Integrovanie telefónie

Telefónia neoddeliteľne patri k službám, ktoré poskytuje UC. Komunikácia hlasom je stále najbežnejšou a najprirodzenejšou formou komunikácie. Od prvého telefonátu po dnešné

formy prenosu hlasu prešla hlasová komunikácia dlhou cestou. Vývoj nám priniesol rôzne technológie, ktoré sa starajú o prenos, uskladnenie, manipuláciu a spracovanie hlasu. Do nedávna sme tu mali prenos hlasu pomocou analógových a digitálnych telekomunikačných sietí národných operátorov, dnes po masívnom nástupe internetu do všetkých oblastí nášho života, nastal vysoký dopyt po prenose hlasu pomocou internetu. Momentálne najbežnejšie používaná je štandardizovaná technológia VOIP. Táto technológia umožňuje prenos digitalizovaného hlasu v tele paketov rodiny protokolov UDP/TCP/IP. Najbežnejšia ústredňa využívajúca túto technológiu je otvorené riešenie Asterisk.

### 3.2 Návrh koncepcie

Koncepcia nám dáva analyticky pohľad na to ako fungujú jednotlivé mechanizmy a súbory mechanizmov. V návrhu koncepcie je systém ako celok systematicky rozdelený do rôznych funkčných blokov, v ktorých sú implementované potrebné mechanizmy. Návrh by mal prihliadať na vyššie uvedené požiadavky.



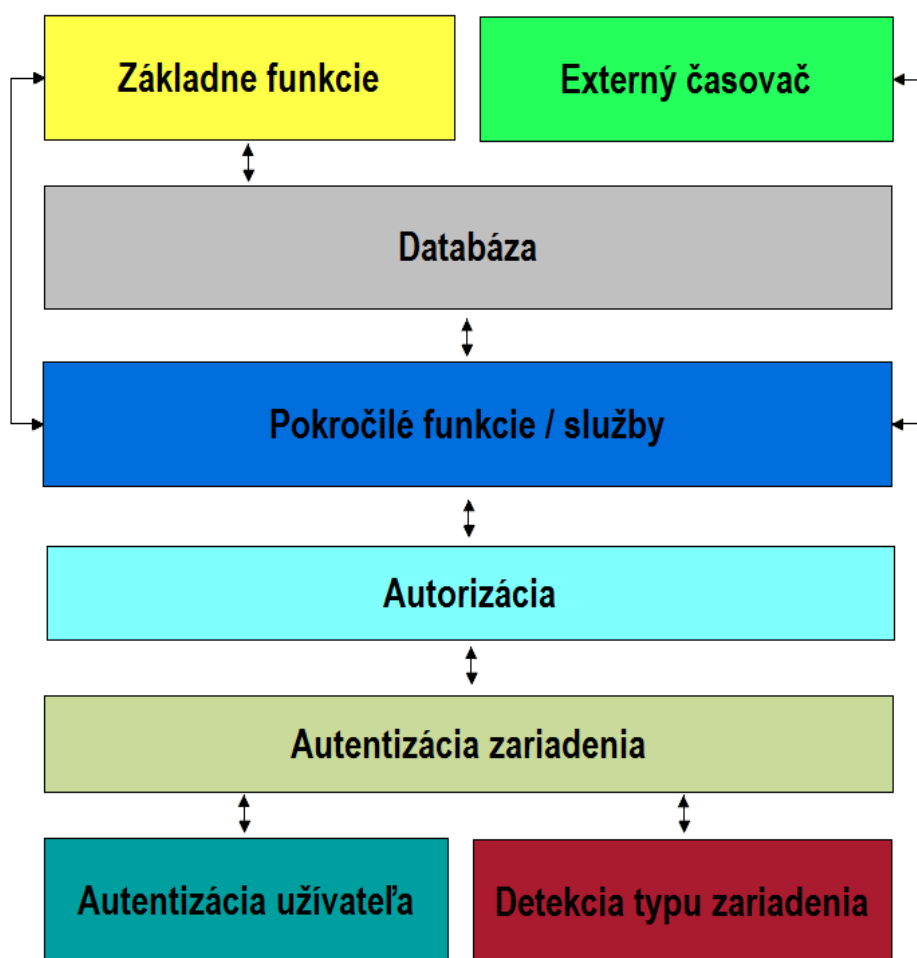
Obrázek 3.1: Schéma hlavného bloku systému

Na obrázku 3.1 je vidieť, že systém je rozdelený do šiestich hlavných blokov. Jadro nám v prvom rade zabezpečuje výmenu informácií medzi jednotlivými blokmi, vykonávanie jednoduchých operácií a prácu s databázou. IM a poštový modul sa stará o manažment jednotlivých implementovaných protokolov, v našom prípade sú to XMPP pre IM a POP3/IMAP/SMTP pre poštové služby, vnútorná poštová komunikácia používa vlastný štandard pre vnútornú komunikáciu. Hlavnou úlohou TTS a hlasového modulu je konverzia rôznych foriem textu na hlas, tento modul by mal dokázať rozpoznávať jazyk textu a pomocou rozhodovacích algoritmov určiť, akým spôsobom bude text konvertovaný. Jedinou úlohou tohoto modulu nie je len konvertovanie textových správ, ale aj podpora hlasových služieb pre front end

a ďalšie pokročilé služby, ktoré nám uľahčujú prístup a používanie UC systému, hlavne pre nevidiacich ľudí pre ktorých je hlas jeden z hlavných dorozumievacích nástrojov komunikácie. Telefónia je jeden z hlavných spôsobov rýchlej komunikácie. Existujú rôzne formy prenášania hlasovej komunikácie, nás hlavne zaujíma prenášanie hlasu pomocou telefónnej ústredne asterisk, ktorá je voľne dostupná pre bežných užívateľov. Hlavnou úlohou modulu telefónie, je integrovanie služieb ústredne do systému UC a opačne. Užívateľské rozhranie nám poskytne pracovné prostredie pre využívanie služieb UC. Modul užívateľského rozhrania by mal podporovať prístup pomocou webového rozhrania a aplikačného rozhrania, ktoré umožní prístup pre špecializované aplikácie, určené pre koncové zariadenia.

### 3.2.1 Jadro platformy

Hlavnou úlohou jadra systému je zabezpečovať komunikáciu medzi jednotlivými modulmi, ďalej má poskytovať služby obsluhy databázy a ďalšie funkcie potrebné pre funkčnosť systému. Vzťahy medzi blokmi v module nájdeme na obrázku 3.2



Obrázek 3.2: Schéma jadra systému

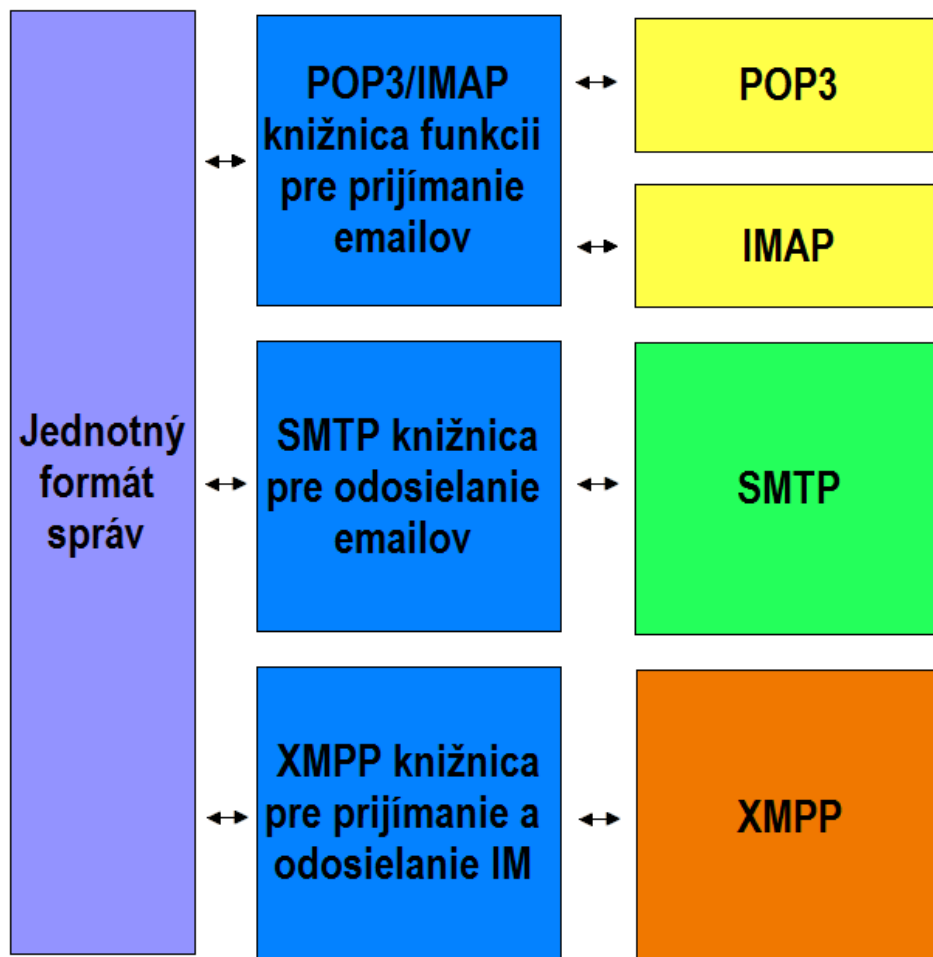
Modul jadra obsahuje množstvo menších funkčných blokov, systematicky rozdelených do presne definovaných skupín. Tento modul musí obsahovať minimálne osem blokov, sú to:

- **Základne funkcie** - V tomto bloku sú zhromaždené jednoduché funkcie, ktoré sú potrebné na fungovanie rôznych systémových funkcií, môžu to byť napríklad operácie s databázou, generovanie potrebných premenných a rôzne ďalšie funkcie.
- **Externý časovač** - Preto aby funkcie mohli vykonávať to, na čo sú určené, ich musí v prvom rade volať objekt, ktorý je na to určený, v našom prípade to môže byť užívateľ, ktorý požaduje prístup k určitej službe alebo to môže byť systémová funkcia, ktorá musí vykonať operácie potrebné pre funkčnosť systému. Preto pre tieto systémové funkcie potrebujeme nezávisle objekty, ktoré budú v určitých časových intervaloch volať presne definované procesy.
- **Databáza** - Je hlavným úložiskom vygenerovaných, získaných a spracovaných dát, ktoré tu môžu byť uložené dočasne alebo dlhodobo. Životný cyklus údajov, dát, správ a iných informácií určujú presne definované požiadavky.
- **Pokročilé funkcie / služby** - Je to súbor pokročilých funkcií a mechanizmov, ktoré na svoj beh potrebujú používať aj funkcie z bloku základných funkcií. Pokročilé funkcie môžu vykonávať pokročilé operácie s dátami uloženými v databáze. Objekty, ktoré chcú využiť služby pokročilých funkcií musia mať na to patričné oprávnenie, toto poverenie musí byť pridelené podľa presne definovaných pravidiel.
- **Autorizácia** - Modul autorizácie prideluje a overuje oprávnenia objektov, ktoré požadujú vykonanie požadovanej služby. Mechanizmus, ktorý obsahuje tento modul musí jednoznačne prideliť oprávnenia objektu podľa dopredu definovaných pravidiel. Taktiež, tento mechanizmus musí zabezpečiť to, že objekt ktorý požaduje poskytnutie konkrétnej služby ma oprávnenie na použitie tejto služby.
- **Autentizácia zariadenia** - Zabezpečuje to, že systém dokáže rozoznať používané koncové zariadenia, pomocou ktorých pristupuje užívateľ k službám, ktoré poskytujú UC. Mechanizmus musí získať informácie o tom, či toto zariadenie už minulosti žiadalo o poskytnutie požadovaných služieb alebo žiada prvý krát. Ďalej to môžu byť IP adresa, typ operačného systému alebo presný čas, kedy zariadenie žiadalo o poskytnutie rôznych služieb.
- **Autentizácia užívateľa** - Pomocou identifikačného mena a hesla nám tento blok overí to, že mám oprávnenie pristupovať k dátam toho užívateľa, za ktorého sa prezentujem. Informácie o užívateľovi, za ktorého sa prezentujem a informácie o zariadení pomocou ktorého pristupujem tvoria objekt, ktorý je potrebný k tomu aby nám systém mohol sprístupniť požadované služby.
- **Detekcia typu zariadenia** - Každé koncové prístupové zariadenie má svoje špecifiká. Tento mechanizmus musí rozoznať, či sa jedná o notebook, tablet alebo inteligentný telefón, preto, aby systém dokázal prispôbiť formu poskytovanej služby pre konkrétne koncové zariadenie.



### 3.2.2 IM a poštový modul

Prijímanie a odosielanie poštových správ pomocou emailu a IM má na starosť tento modul. Hlavnou úlohou modulu je komunikácia s protokolmi POP3/IMAP/SMTP a XMPP, odkiaľ prijíma a odosiela správy. Prijaté správy transformuje do takej formy, ktorá spĺňa požiadavky jednotného formátu, tu ukladá informácie o odosielateľovi správy, predmet, text správy, dáta a čas prijatia a iné dôležité informácie. Emailové správy prijíma pomocou protokolu POP3 a IMAP, a pomocou SMTP správy odosiela. IM pre prijímanie a odosielanie rýchlych správ používa iba XMPP protokol. XMPP dokáže nielen vymieňať správy, ale má v sebe implementované rôzne ďalšie funkcie, ako je napríklad získanie momentálneho stavu užívateľa, to znamená, že dokáže určiť či je užívateľ momentálne dostupný. Pre implementovanie tohoto modulu použije už existujúce knižnice, kde sú súbory funkcií, ktoré pracujú s týmito protokolmi. Tieto knižnice majú k dispozícii vyššie programovacie jazyky, preto nám odpadá nutnosť implementovať vlastné riešenia komunikácie medzi systémom a protokolmi.



Obrázek 3.3: *Bloková schéma poštovného modulu*

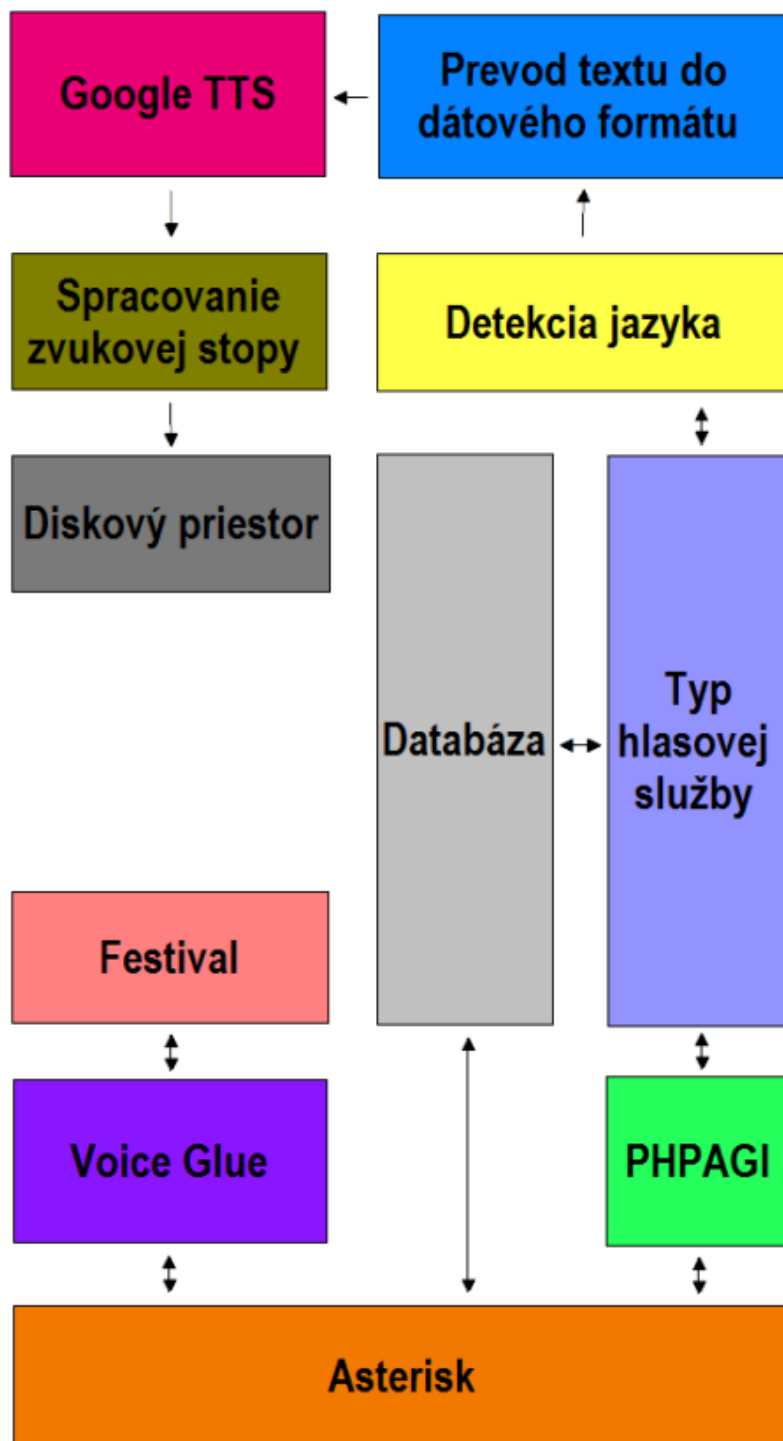
Prijaté správy transformuje knižnica do dátového poľa, ktoré potom zoberie blok jednotných správ a ten potom analyzuje, získané dáta pomocou jadra systému uloží na konkrétne

miesto v databáze. Náš implementovaný poštový modul je zobrazený v blokovej schéme na obrázku 3.3, bude obsahovať minimálne štyri bloky, sú to:

- Jednotný formát správ - Mechanizmus jednotného formátu musí transformovať formu dátového poľa, ktoré nesie dáta o správe do formátu, ktorý splňuje požadovane parametre. Blok musí zvládnuť transformáciu dátového poľa správy do jednotného formátu alebo z jednotného formátu do formátu určeného pre rôzne komunikačne protokoly.
- POP3/IMAP knižnica funkcií pre prijímanie emailov - Je to súbor funkcií a mechanizmov, ktoré sa starajú o nadviazanie spojenia s poštovým serverom, získavanie informácií o jednotlivých správach a sťahovanie týchto správ do lokálnej premennej. Ďalej, obsahuje mechanizmy, ktoré sa starajú o zabezpečenie komunikácie alebo o prekódovanie textu do inej znakovkej sady.
- SMTP knižnica pre odosielanie emailov - Táto knižnica je opakom tej predchádzajúcej. Stará sa o to aby presne definované dátové pole bolo odoslané ako emailová sprava. Taktiež podporuje funkcie zabezpečenia.
- XMPP knižnica pre prijímanie a odosielanie IM - Pre rýchle IM správy sa používa iba jediný protokol XMPP, ktorý slúži na prijímanie a odosielanie správ, získavanie informácií o stave užívateľa, synchronizácia uložených kontaktov na servery. Tento modul reálne vystupuje ako klientská aplikácia, tým pádom nemusíme prevádzkovať serverovú časť.

### 3.2.3 TTS a hlasový modul

Jeden z dôležitých modulov pre UC, jeho funkciou je preklad správ z textovej formy do formy hlasu. Modul by mal prekladať nielen správy ale aj rôzne texty pre rozšírené funkcie užívateľského rozhrania. Vytvorené hlasové stopy by mali byť k dispozícii aj pre iné funkčné celky systému ako je napríklad telefónna ústredná asterisk. Mechanizmus modulu by mal byť pripravený na ďalšie systémy TTS ktoré sa budú implementovať v budúcnosti. Momentálne máme k dispozícii dva hlasové systémy. Je to hlasový systém od spoločnosti Google a otvorené riešenie prekladu na hlas ktoré sa nazýva Festival. Riešenie Festival je súčasťou balička, ktorý je poskytovaný pre ústrednú asterisk. Náš hlasový modul bude podporovať oba dva moduly. Podporu hlasu pomocou služieb poskytovaných od spoločnosti Google ktorá ma vyššiu kvalitu spracovania hlasu budeme preferovať pred otvoreným riešením Festival. K službám od spoločnosti Google môžeme pristupovať rôznymi spôsobmi, nás hlavne bude zaujímať prístup pomocou oficiálneho API alebo pomocou webových formulárov. Pre lepšie spracovanie textu na hlas je detekcia jazyka dôležitá preto aby algoritmus dokázal lepšie aplikovať potrebné mechanizmy. Tuto detekciu jazyka nám ponúkajú služby od Googlu. Detekcia od tejto spoločnosti ma vysokú pravdepodobnosť určenia správneho jazyka. Pre transformáciu textu na hlas pomocou Festivalu musíme požiadavky interpretovať pomocou Voice xml jazyka. Dôležité bloky modulu sú zobrazené v schéme na obrázku 3.4.



Obrázek 3.4: Bloková schéma hlasového modulu

- Typ hlasovej služby - Je to algoritmus, ktorý rozhoduje o tom, pomocou ktorého prekladača bude spracovaná požiadavka na vytvorenie zvukovej stopy. Podľa typu požadovanej služby vyhodnotí ktorý prekladač je vhodnejší na spracovanie hlasu.

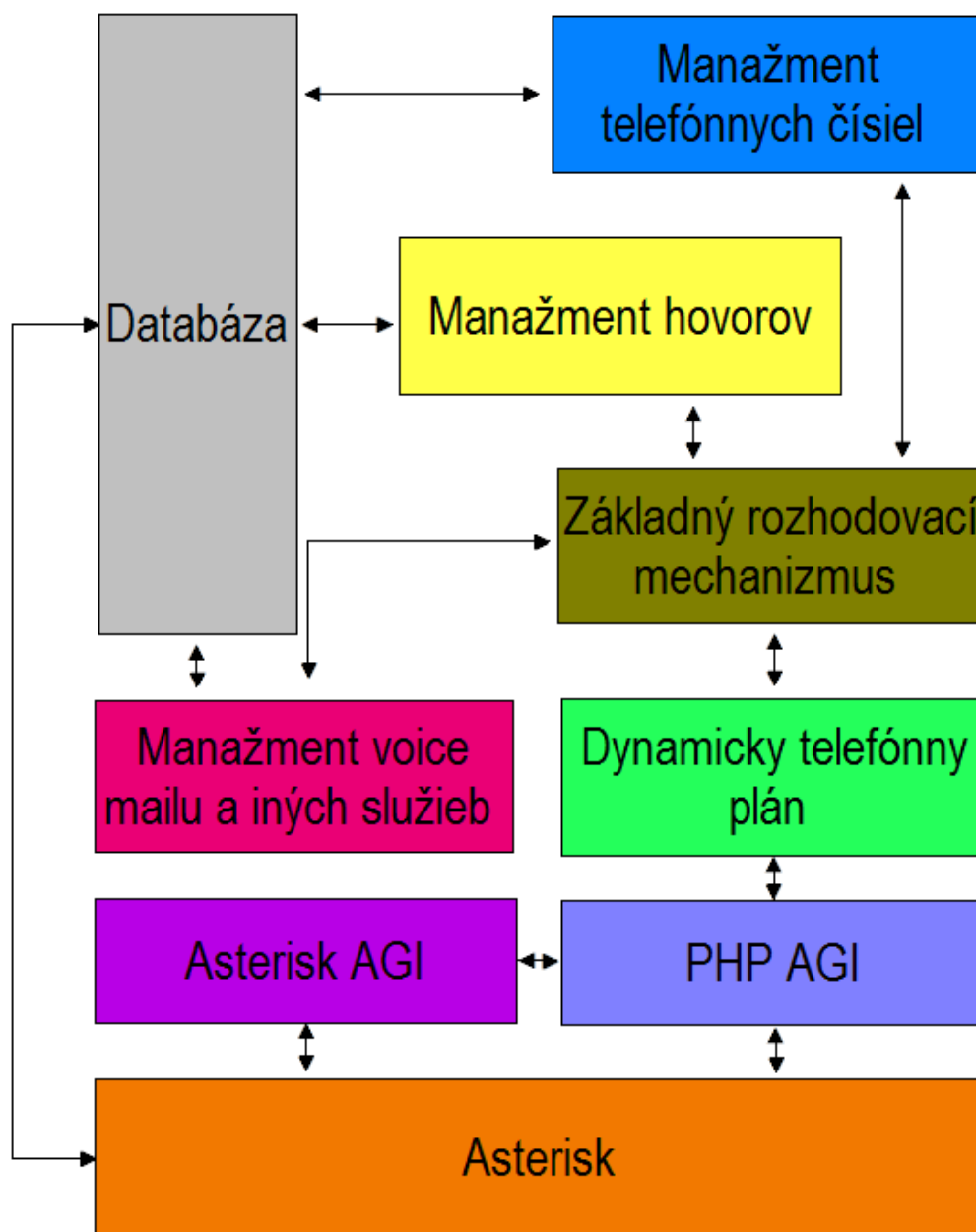
Modul by mal taktiež obsahovať priebežnú kontrolu spracovania hlasu, ktorá nám povie v akom štádiu sa nachádza požiadavka na spracovanie hlasu.

- Detekcia jazyka - Pre uľahčenie práce užívateľom je automatická detekcia jazyka najlepšia voľba, ale na túto detekciu treba pokročilý systém, ktorý má vysokú efektivitu detekcie, preto by náš systém mal ponúkať aj manuálnu voľbu. Modul by mal podporovať jednoduchú implementáciu nového systému rozpoznávania jazyka a možnosť kombinácie viacerých systémov.
- Prevod textu do dátového formátu - Tento blok modulu sformuluje požiadavku, ktorá sa má odoslať na komunikačné rozhranie služby TTS od Googlu. Tento mechanizmus zabezpečí transformáciu prideleného textu do takej formy, ktorá spĺňa požiadavky pre odoslanie a následovne spracovanie. Blok musí zmeniť kódovaciu sadu tak, aby celistvosť a zrozumiteľnosť textu zostala zachovaná.
- Google TTS - Je to služba prekladu textu na hlas od spoločnosti Google. Blok v našom modeli môžeme chápať ako vstupno-výstupné rozhranie pre zaslanie požiadavky na preloženie textu. Toto rozhranie prijíma požiadavky, ktoré majú presne definovanú formu. Po prijatí a spracovaní požiadavky nám poskytne zvukovú stopu.
- Spracovanie zvukovej stopy - Vytvorenie hlasovej zvukovej stopy musíme určitým mechanizmom prevziať a uložiť na presne určené miesto v našom systéme. O každej vytvorenej zvukovej stope vedíme záznam v databáze, čo nám dáva prehľad o tom, aké texty boli prevedené na text alebo jednu zvukovú stopu, môže používať viacero užívateľov a iné ďalšie možnosti.
- Diskový priestor - Je to presne vyhradené miesto v pamäti systému, kde sa ukladajú výlučne zvukové stopy. Každá zvuková stopa má pridelený jedinečný identifikátor a používa štandardný zvukový formát, ktorý má priaznivý pomer kvality a veľkosti.
- Festival - Je rozšírenie hlasových služieb pre ústrednú asterisk. Využívať služby tohoto rozšírenia môžeme pomocou interpretačného jazyka Voice XML. Toto rozšírenie potrebuje pre svoju funkčnosť rôzne ďalšie balíky a aplikácie. Požiadavky na toto rozšírenie neposielame priamo, ale pomocou rôznych interpreterov. Najpoužívanejším otvoreným riešením je Voice glue.

### 3.2.4 Integrácia telefónnych služieb

Hlasová komunikácia patrí k najpoužívanejšej forme komunikácie, má veľa výhod, ale aj problémov. Hlbšia integrácia telefónnych služieb nám umožní riešenie niektorých z týchto problémov. Dá nám to nové možnosti a pohľad na telefónnu komunikáciu. Telefonovanie sa stáva súčasťou komunikačného balíka, ktorý nám umožní lepši manažment telefónnych čísel a hovorov. Nové služby, ktoré vzniknú, sú napríklad, jednotné telefónne číslo pre viac používaných zariadení, ďalej to môže byť presmerovanie telefónnych hovorov v prípade nedostupnosti. Systém môže ukladať telefónne záznamy o uskutočnených hovoroch. Dostupnosť telefónnych čísel môžeme určovať podľa časového hľadiska, čo znamená, že číslo môže byť dostupné v určitých časových intervaloch a mnoho ďalších užitočných služieb. Tvorba telefónneho plánu sa presúva z ústredne na rozhodovací mechanizmus UC systému. Pre každého súčasníka sa generuje

dynamicky v prípade potreby. Na obrázku 3.5 je súbor blokov ktoré potrebujeme pre funkčnosť tohoto modulu.



Obrázek 3.5: *Bloková schéma telefónneho modulu*

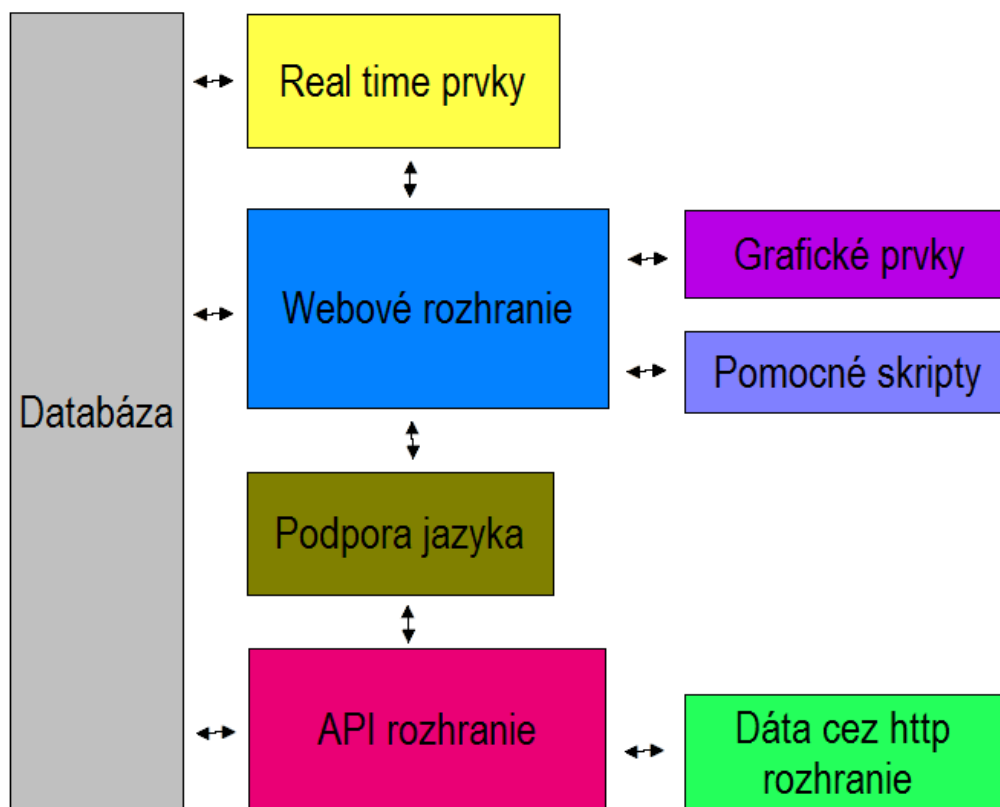
- **Základný rozhodovací mechanizmus** - Rozhoduje o tom ako spracuje vstupne dáta alebo požiadavky. Poskytuje rôzne potrebné informácie pre blok dynamicky generovaného plánu a spracováva informácie, ktoré mu tento plán poskytol.
- **Dynamický telefónny plán** - Je to súbor algoritmov, ktoré generujú dynamický telefónny plán pre každého účastníka samostatne. Algoritmus sa rozhoduje podľa viacerých podmienok, ktoré definoval užívateľ podľa vlastných preferencií.

Vytvorený telefónny plán potom posíla do bloku PHP AGI ktorý aplikuje priradený plán. Dynamicky plán by mal byť kompatibilný s rôznymi verziami asterisku, kde dochádzalo k priebežným zmenám vo verziách.

- Manažment telefónnych čísiel - Je to blok, ktorý sa stará o manažment telefónnych čísiel. Pomocou neho môže užívateľ pridávať, odstraňovať alebo upravovať telefónne čísla. Tiež poskytuje rôzne potrebné informácie pre vytvorenie dynamického plánu ako napríklad či je číslo momentálne aktívne alebo na aké ďalšie číslo má presmerovať hovor.
- Manažment hovorov - Poskytuje informácie o prebiehajúcich a uskutočnených hovoroch. Spracováva uložené telefónne záznamy a priradzuje ich ku konkrétnym užívateľským účtom.
- Manažment voice mailu a iných služieb - Vytvára pracovné prostredie pre tvorbu voice mailu pomocou telefónu alebo počítača. Môže spravovať ďalšie služby, ktoré budú implementované v budúcnosti.
- PHPAGI - Poskytuje nám rozhranie na ovládanie funkcií a vnútorných mechanizmov asterisku. Interpretuje skriptovací jazyk PHP do formy pomocou ktorej riadi ústredňu. Pomocou tohoto rozhrania aplikujeme dynamicky telefónny plán a asterisk spätne poskytuje rôzne informácie o stave ústredne.

### 3.2.5 Užívateľské rozhranie

Prístupové užívateľské rozhranie je dôležitým prvkom nášho systému pretože vytvára grafické pracovné prostredie potrebné pre využívanie služieb, ktoré tento systém ponúka. Grafické rozhranie musí spĺňať niekoľko požiadaviek na to aby toto rozhranie bolo prívetivé, flexibilné, univerzálne, aby zvyšovalo efektivitu práce, šetrilo čas a doručovalo správy tým najvhodnejším spôsobom. Toto prostredie by malo sledovať moderné trendy a využívať najnovšie dostupné technológie. V našom prípade budeme používať webové rozhranie a taktiež by systém mal ponúkať možnosť prístupu k dátam pomocou rôznych aplikácií ktoré sú integrované do rôznych koncových zariadení. Grafické webové rozhranie by malo používať 3D grafické prvky, real time prvky, ktoré urýchľujú a zjednodušujú prácu s týmto rozhraním. Rozhranie ako celok by malo byť optimalizované pre rôzne druhy zariadenia a čo najmenej vyťažovať konektivitu. Užívateľ by mal mať možnosť si prispôbiť grafické prvky rozhrania podľa vlastných preferencií. Jazyková podpora by mala byť oddelená od systému preto aby budúca implementácia nového jazyka nezasahovala do funkcií a behu systému. Lepšiu predstavu aké funkčné bloky systém potrebuje, nám dáva schéma modulu na obrázku č.3.6.



Obrázek 3.6: *Bloková schéma modulu webového rozhrania*

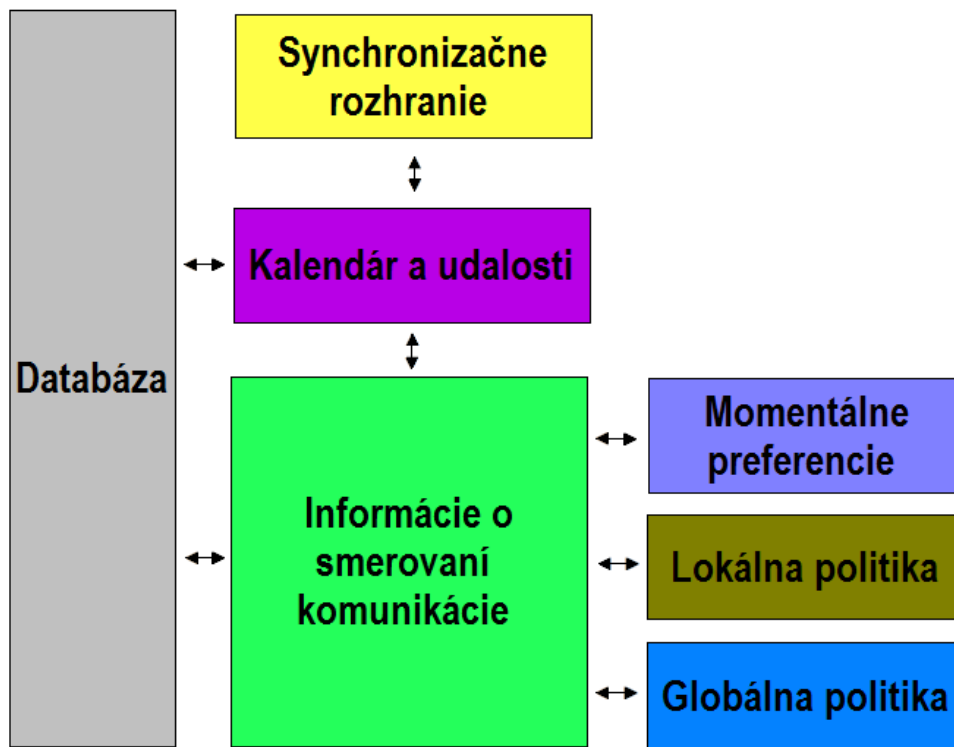
- **Webové rozhranie** - Tento modul v sebe ukladá presne definované grafické štruktúry ako sú tabuľky, objekty, bloky a iné ďalšie prvky. Štruktúra obsahu by mala byť optimalizovaná pre rôzne druhy prehliadačov a rozlíšení zobrazovacieho média. Implementácia tejto štruktúry musí mať jednotné pravidla pre všetky použité prvky aby dynamická tvorba obsahu bola čo najefektívnejšia z technologického pohľadu.
- **Real time prvky** - Hlavnou úlohou je dať rozhraniu určitú dynamiku, ktorá zabezpečí jednoduchšiu, intuitívnejšiu a rýchlejšiu prácu s rozhraním. Medzi najpoužívannejšie prvky patri automatická aktualizácia obsahu, automatické dopĺňovanie rôznych formulárov, automatické oznámenie doručenia novej správy alebo udianie sa novej skutočnosti.
- **Podpora jazyka** - Ako sme už uvádzali tak jazyková podpora by mala byť oddelená od systému pre rôzne výhody. Pre nás sú najdôležitejšie jednoduchá implementácia a možno nezávislého používania pre rôzne zariadenia. Databáza slov by mala mať formu, ktorá dovoľí jednoduchú úpravu a doplnenie rôznych slov a výrazov.
- **API rozhranie** - Je pravdepodobne že v budúcnosti by mohli existovať aplikácie vytvorené pre zariadenia s operačným systémom Android, Windows mobile alebo pre IOS. Preto by sme mali pri návrhu zahliadť aj tento faktor a predprípraviť systém na budúcu implementáciu.

- Dáta cez http rozhranie - Aplikáciou v telefóne môže pristupovať cez rozhranie API. Je to rodina Hypertextových prenosových protokolov (HTTP) s XML serializáciou v spojení s ďalšími súvisiacimi webovými štandardmi. Pomocou neho si môže aplikácia vymieňať potrebné dáta.
- Grafické prvky - Celková grafika systému by sa mala držať moderných trendov ako sú plynulý prechod a načítavanie v pozadí. Jednoduché rýchle animácie a 3D grafické prvky s vysokým rozlíšením, ale súčasne musí byť zachovaná pružnosť systému a minimálna veľkosť dát potrebných na načítanie obsahu.
- Pomocné skripty - Časti webového rozhrania budú implementované s použitím rôznych technológií ako sú CSS, Java script, AJAX, jQuery a ďalšie. Pomocou kombinácie rôznych technológií docielime rôzne pozitívne schopnosti pri práci s rozhraním. Pracovné prostredie bude interaktívnejšie a viac užívateľsky prívetivé. V tomto bloku budú uložené časti kódov.

### 3.2.6 Modul smerovania komunikácie

Doposiaľ sme sa zaoberali tým ako získavať správy z komunikačných kanálov, ako ich prekladať na hlas alebo ako ich poskytnúť užívateľovi, ale musíme si uvedomiť že tu existuje aj časový faktor. Spôsob a potreba komunikácie sa mení v čase. Užívateľ nemusí mať vždy možnosť prijať hovor alebo email preto systém musí prihliadať na to aké možnosti komunikácie má užívateľ momentálne k dispozícii. Máme niekoľko možností ako zistiť momentálne možnosti užívateľa, je tu možnosť automatickej detekcie systémom, kde systém určí potencionálne možnosti a podľa určitých pravidiel rozhodne ako bude zaobchádzať s prijatou komunikáciou. Ďalšia možnosť je, že systému dáme možnosť prístupu ku kalendáru a udalostiam. Podľa týchto informácií rozhodne a ďalšom postupe. Informáciu z kalendáru bude získavať synchronizáciou z iných bežne používaných kalendárových aplikácií ako napríklad MS Outlook. Aplikovaním rôznych pravidiel komunikácie zabezpečíme najhodnejší prístup a spracovanie užívateľskej komunikácie. O tom akým spôsobom sa bude spracovávať komunikácie rozhodnú pravidla rozdelené do troch kategórií, sú to momentálne preferencie, lokálna užívateľská politika a globálna firemná politika. Na obrázku 3.7. môžem nahliadnuť pod povrchu modulu smerovania komunikácie.





Obrázek 3.7: *Bloková schéma modulu smerovania komunikácie*

- Synchronizačne rozhranie - Spracováva dáta vo forme rôznych výstupov kalendárnych aplikácie, ktoré vyexportujú časový harmonogram užívateľa. Získané dáta na vstupe rozhrania importuje v určitej forme do databázy nášho systému.
- Kalendár a udalosti - Časový harmonogram si môžem vytvoriť aj cez webové zohranie, kde môžem spravovať udalosti. Sem budú zobrazene aj udalosti, ktoré boli importované z inej externej aplikácie. Hlavnú činnosť, ktorú tento modul vykonáva je, že poskytuje smerovaciemu mechanizmu informácie o momentálnom stave užívateľa. Momentálny stav môžem definovať aj pre komunikačné kanály, kde môžeme definovať kedy bude komunikácia pomocou tohto kanála aktívna alebo akým spôsobom má spracovávať komunikáciu z tohoto kanálu v závislosti od času.
- Smerovanie komunikácie - Má na starosť aplikovať jednotlivé pravidlá pre komunikáciu v závislosti na čase. Posudzuje, ktoré pravidla budú aplikované pre konkrétny komunikačný kanál a ktoré z tých pravidiel ma vyššiu prioritu.
- Momentálne preferencie - Sú to momentálne užívateľské preferencie, ktoré majú najvyššiu prioritu. Sú väčšinou uplatňované v krátkych časových intervaloch. Väčšinou sa aplikujú pri nečakaných zmenách v časovom harmonograme.
- Lokálna politika - Lokálna užívateľská politika ma vyššiu prioritu ako globálna firemná politika, tato priorita neplatí v prípade keď je globálna politika vynútená. Tato politika sa aplikuje zväčšia na dlhšie časové obdobie. Pravidla politiky si definuje každý užívateľ zvlášť podľa vlastných preferencií.

- Globálna politika - Globálna firemná politika ma najnižšiu prioritu, to neplatí v prípade ak je kvôli určitým dôvodom globálne vynútená. Za normálnych podmienok definuje pravidlá komunikácie pre všetkých užívateľov systému. Je časovo neobmedzená, ale môže byť v určitých prípadoch aj obmedzená.

### 3.3 Vytvorenie rôznych častí platformy

Doposiaľ sme sa zaoberali teoretickým návrhom systému, rozdelili sme ho do rôznych funkčných modulov. Opísali sme spôsob práce blokov v jednotlivých moduloch. Uviedli sme, aký typ dát je potrebný na vstupe jednotlivých blokov a ako budú vyzerat' výstupy týchto blokov. Ďalší krok, ktorý bude nasledovať je implementácia rôznych funkcií a mechanizmov v jednotlivých blokoch. V tejto časti kapitoly budeme navrhovať algoritmy, ktoré budú napísané v skriptovacom jazyku PHP. Rozoberieme si jednotlivé kroky, ktoré musí algoritmus vykonať. V tejto práci nebudeme uvádzať jednotlivé zdrojové kódy, tie budú poskytnuté ako príloha k tejto práci. Uvádzať tu budeme iba hlavné časti kódu, ktoré sme označili ako dôležité funkcie.

#### 3.3.1 Jadro systému

Hlavnou úlohou jadra systému je zabezpečiť komunikáciu medzi systémom a databázou. Vkladá, upravuje a vyberá jednotlivé záznamy, ktoré požaduje systém pre ďalšie spracovanie. Ďalej zabezpečuje správne fungovanie systému ako celku. V prípade poruchy alebo zlyhania tuto udalosť systém zaznamená a upozorní na to poverené osoby. Systém sa snaží detegovať potencionálne hrozby a eliminovať ich.

##### 3.3.1.1 Základne funkcie

Sú to jednoduché operácie vykonávané pre funkčnosť systému, ako napríklad overenie existencie prístupu k databáze alebo funkčnosť potrebných PHP modulov, ktoré nám poskytujú rozšírenie potrebných funkcií, sú to väčšinou knižnice pre jednotlivé protokoly. Môžu to byť funkcie, ktoré overujú existenciu súborov alebo zisťujú základné informácie o servery na ktorom prevádzkujeme služby. V prípade, ak by detekcia dospela k tomu, že funkčnosť systému je narušená, môže vydať pokyn na reaktiváciu rôznych služieb alebo celkové odstavenie systému preto, aby eliminoval vzniknutie rôznych škôd a únik informácií.

##### 3.3.1.2 Úložisko dát

Každý pokročilý systém potrebuje pre svoju funkčnosť úložisko dát, ktoré musí spĺňať určité podmienky, aby bola zaručená určitá rýchlosť prístupu k dátam, minimálne nároky na hardware, maximálna bezpečnosť, integrita dát a jednoduchá obsluha databázy. Pre naše účely budeme používať otvorené riešenie Mysql, ktoré je bezplatne dostupné pre bežných užívateľov. Táto databázová technológia nám poskytne jednoduché a rýchle spracovanie dát, je to časom overená technológia, preto sa nemusíme obávať o bezpečnosť a funkčnosť tohoto produktu. Preto, aby sme mohli používať tento typ databázy musíme ju najskôr nainštalovať na požadovaný server a nakonfigurovať základne parametre, ako je adresa a port pre prístup k rozhraniu, ďalej to je užívateľské meno a heslo administrátora databázy. Po nainštalovaní databázy musíme povoliť rozhranie medzi PHP serverom a touto databázou, v našom prípade musíme povoliť rozšírenie

php\_mysql.dll alebo to môže byť aj php\_mysqli.dll. Po overení funkčnosti prepojenia vytvoríme databázu potrebnú pre náš systém, budeme ju nazývať chronos podľa názvu nášho systému. Aby nám databáza povolila prístup k užívaniu, musíme vytvoriť konfiguračný súbor, ktorý sa bude nazývať config.php. V ňom definujeme nasledovné parametre:

```
< -- Parametre potrebne na pripojenie k databáze. -->
DEFINE ('DB_UZIVATEL', 'Užívateľské meno');
DEFINE ('DB_HESLO', 'Užívateľské heslo');
DEFINE ('DB_HOSTITEL', 'Adresa hostiteľa');
DEFINE ('DB_NAZEVDATABAZE', 'Názov databázy');
< -- Vytvorenie spojenia. -->

$dbc = @mysql_connect (DB_HOSTITEL, DB_UZIVATEL, DB_HESLO)
OR die ('Nemôžem sa pripojiť k databáze: '. mysql_error());

mysql_select_db (DB_NAZEVDATABAZE) OR die ('Nemôžem vybrať
databázu: '. mysql_error());

< -- Definovanie kódovacej sady pomocou ktorej bude prebiehať
komunikácia. -->

MySQL_Query("SET NAMES 'cp1250'");
```

Pre podporu slovenčiny a češtiny budeme používať kódovaciu sadu cp1250, určenú pre región strednej Európy. Ak sa nepodari pripojiť k hostiteľovi alebo k databáze, systém zhlási chybu.

### 3.3.2 Webové grafické rozhranie

V podkapitole návrh sme definovali rôzne požiadavky na vlastnosti, ktoré budú aplikované pri implementácii webového rozhrania. Pracovné prostredie sa bude skladať z troch častí, medzi nimi sa budeme premiestňovať plynulým prechodom pomocou grafických prvkov alebo pomocou skrolovania tlačidlom na myši. Veľkosť každej časti bude závisieť od rozlíšenia obrazovky. Systém deteguje zobrazovaciu veľkosť obrazovky a prispôbi veľkosť každej časti. Tieto tri časti pracovného prostredia budeme nazývať Domov, Kalendár a Hub. Pre lepšie pochopenie je na obrázku 3.8. zobrazené pracovné prostredie ktoré je rozdelené do troch častí.

- Časť domov - Je to úvodná stránka, kde sa nám zobrazujú základné informácie o užívateľskom účte. Môžeme tu upravovať základné informácie o užívateľovi a prispôbiť grafické prvky podľa preferencií pre každého užívateľa zvlášť. Ďalej tu môžeme vytvárať pravidla pre lokálnu politiku alebo nastaviť momentálnu preferenciu.

- Časť kalendár - V tejto časti na nachádza kalendár, v ktorom bude zobrazovaný časový harmonogram užívateľa, ktorý bol vytvorený pomocou tohoto rozhrania alebo bol synchronizovaný z externej kalendárovej aplikácie. Môžeme tu pridávať rôzne upomienky a iné časové udalosti.
- Časť hub - Hub je miesto kde sú zobrazené prijaté správy zo všetkých komunikačných protokolov. Sem ich môžeme čítať, upravovať alebo preposielať pomocou iného protokolu. Ku každej správe si môžeme dať vytvoriť hlasovú nahrávku prijatej správy a nasledovne si ju vypočuť pomocou poskytnutého zvukového prehrávača.



Obrázek 3.8: Rozdelenie pracovného prostredia

### 3.3.2.1 Použitá technológia

Vo webovom rozhraní sú na rôzne časti použité rôzne technológie. Štruktúra obsahu je napísaná pomocou jazyka xhtml, ktorý nám zabezpečí kompatibilitu pre rôzne zariadenia. Spôsob akým sa budú zobrazovať stránky napísané v xhtml je definovaný pomocou kaskádových štýlov,

ponuka nám možnosť oddeliť vzhľad od štruktúry a obsahu stránky. V našom prípade používame rozšírenú verziu CSS 2. Pre jednoduché interaktívne prvky budeme používať JavaScript, ktorý budeme vkladať do xhtml štruktúry. Real time prvky budú fungovať na baza technológie AJAX, ktorá nám zabezpečí napríklad automatické aktualizovanie obsahu alebo rôzne grafické efekty. Pre pokročilé interaktívne prvky ako je predikcia textu nám poslúži technológia JQuery, je to knižnica funkcií, ktoré sú napísané JavaScriptu a ďalších technológií. Kombináciou týchto technológií dosiahneme vysokú interaktivitu, rýchlu odozvu a intuitívne ovládanie.

### 3.3.2.2 *Modularita*

Modularita systému by mala byť implementovať na každej úrovni systému. V našom prípade je modularita aplikovaná v týchto prípadoch:

- Spracovanie dát je oddelene od štruktúry - Všetky mechanizmy a funkcie potrebné pre spracovanie dát sú oddelené od štruktúry webového rozhrania. Všetky tieto skripty sa nachádzajú v samostatných súboroch rozdelených podľa pracovného účelu. Premenné, ktoré vytvorila skriptovacia časť systému sú po dokončení vložené do štruktúry stránky. Štruktúra webového rozhrania sa nachádza v priečinku s názvom template.
- Grafika je oddelená od štruktúry a obsahu - Grafické prvky ako sú farby, typ a veľkosť písma, grafika pozadia, veľkosť a štýl tabuliek alebo forma ovládacích prvkov je definovaná pomocou kaskádových štýlov. Tieto štýly sú rozdelené do rôznych súborov rozdelených podľa účelu. Všetky súbory obsahujúce kaskádové štýly sú umiestene v priečinku s názvom style. Takto definovaná grafika rozhrania nám redukuje množstvo kódu ktorý je potrebný pre správne zobrazenie obsahu.
- Štruktúra je oddelená od obsahu - Pre dynamickú tvorbu obsahu musí byť štruktúra oddelená od obsahu. Obsah je do štruktúry vkladán pomocou lokálnych premenných, ktoré napĺňame dátami podľa potreby. To znamená, že jedna štruktúra môže zobrazovať rôzne druhy obsahu bez toho aby bola tato štruktúra nejako ovplyvnená týmto obsahom.
- Interaktívne prvky sú oddelene od štruktúry - Mechanizmy pre tvorbu interaktívnych prvkov sú umiestené vo vlastných súboroch rozdelených podľa definovaného účelu. Zjednodušuje to implementáciu nových prvkov. V prípade nekompatibility prvkov a zariadenia s ktorým tieto prvky spolupracujú môže systém jednoducho vymeniť mechanizmus tvorby takéhoto prvku za kompatibilný. Súbory týchto prvkov sú umiestnené v priečinku script.

### 3.3.2.3 *Podpora jazyka*

Podpora rôznych druhov jazykov je dnes už samozrejmosť. V našom prípade je jazykový obsah oddelený od štruktúry webového rozhrania. V tejto štruktúre sú umiestnené lokálne premenné, ktoré si nahrávajú dáta z externého súboru. Tieto súbory sú umiestene v priečinku s názvom lang ktorý sa ďalej delí do ďalších pod priečinkov ktoré sú pomenované podľa skratiek jednotlivých jazykov, napríklad angličtina má názov en. Výber jazyka závisí od rôznych faktorov. V prvom rade si systém overí či prístupové zariadenie už niekedy v minulosti pristupovalo na



webové rozhranie, ak pristupovalo tak zvoli jazyk aký bol používaný naposledy, ak zariadenia pristupuje prvý krát tak zvoli jazyk ktorý je určený ako predvolený. Po prihlásení užívateľa sa zvoli nim preferovaný jazyk. V prípade ak chceme implementovať nový jazyk vytvoríme priečinok, doňho vložíme súbor s jazykovými podkladmi a upravíme nastavenia v knižnici funkcií systému ktorá sa nazýva features.php. V tejto knižnici sa nachádzajú tri funkcie pre potrebu určenia a výberu vhodného jazyka, sú to:

- default\_language() - Získava informáciu o predvolenom jazyku
- select\_lang() - Určuje či rozhranie bude používať predvolený jazyk alebo jazyk, ktorý bol používaný v minulosti.
- select\_language() - Tato funkcia načíta patričný súbor s jazykovými podkladmi.

### 3.3.2.4 Rozmiestnenie informačných a príkazových prvkov

Rozmiestnenie aktívnych a pasívnych prvkov je intuitívne. V ľavom hornom rohu sa nachádza ikona návratu. Pod ňou sa nachádza hlavné menu. V strede stránky sa nachádza hlavné pracovné prostredie a nad týmto prostredím sa nachádza takzvaný command bar. V pravom hornom rohu sa nachádza prvok rýchleho prístupu. Pre lepšiu predstavu je toto rozmiestnenie zobrazené na obrázku 3.9.



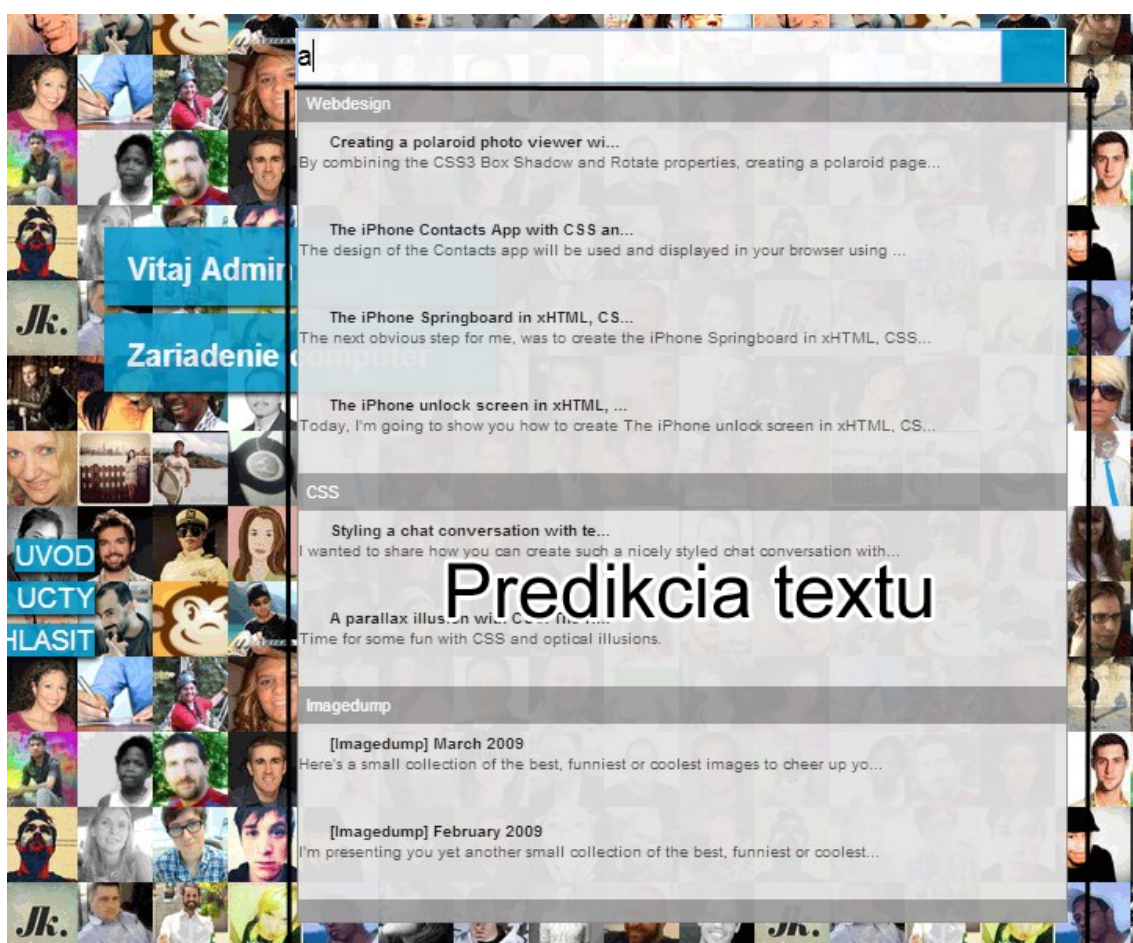
Obrázek 3.9: Rozmiestnenie informačných a príkazových prvkov

- Hlavná pracovná plocha - Ako vyplýva z názvu je to hlavná plocha kde sa zobrazuje hlavný obsah. Tento obsah sa dynamicky aktualizuje podľa požiadaviek užívateľa. Na aktualizáciu netreba znovu načítať celú stránku.
- Command bar - V preklade to znamená príkazový riadok, je to interaktívny prvok, ktorý slúži na vyhľadávanie alebo vykonávanie zadaných výrazov. Pomocou kľúčových slov môže vykonávať rozličné operácie. Tento príkazový riadok disponuje predikciou textu.
- Hlavne menu - Nachádzajú sa tu odkazy pre jednotlivé funkcie systému. Po aktivácii odkazu sa načíta požadovaný obsah do pracovného prostredia. Nachádzajú sa tu odkazy pre nastavenie užívateľského účtu, lokálna politika a rôzne ďalšie.

- Ikona návratu - Tento prvok slúži na návrat do úvodného obsahu. Ikona sa pohybuje nezávisle od obsahu, vždy je umiestnená na rovnakom mieste.
- Rýchly prístup - Slúži na rýchly prechod medzi domovom, hubom a kalendárom. Tento prvok sa taktiež pohybuje nezávisle od obsahu. Obsah tohoto prvku sa môže meniť v závislosti od miesta kde sa momentálne užívateľ nachádza.

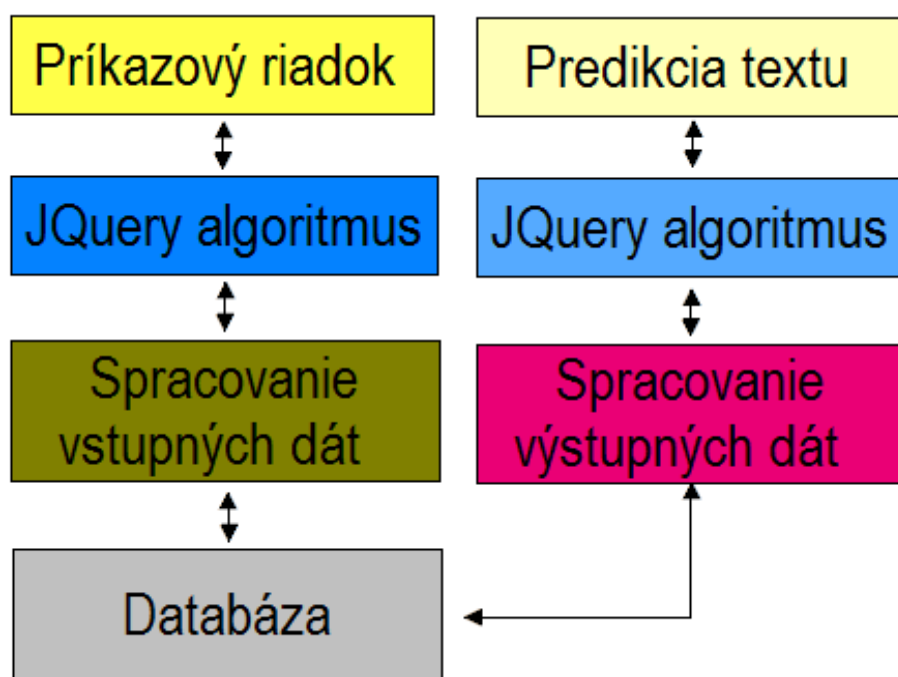
### 3.3.2.5 Prvky pre zrýchlený prístup

Je to kľúčový prvok pre rýchle získavanie informácií alebo rýchle spracovanie informácií. Pomocou tohoto príkazového riadku sa môžeme expresne dostať k potrebným informáciám alebo vykonať požadovanú operáciu. Predikcia textu nám umožňuje intuitívnejšie ovládanie príkazového riadku. Ako je vidno na obrázku 3.10 po zadaní znaku začne automatické vyhľadávanie.



Obrázek 3.10: *Predikcia textu*

Mechanizmus príkazového riadku pracuje na baze JQuery a PHP. Algoritmus napísaný v jQuery automaticky posiela zadané znaky mechanizmu napísaného v PHP ktorý upraví vstupné dáta do formy, ktorú nasledovne žiada potrebné informácie z databázy. Výstupné dáta z databázy sa nasledovne spracujú do konečnej formy ktorá bude okamžite zobrazená vo forme predikcie textu. Nasledovný postup je zobrazený pomocou diagramu na obrázku 3.11.

Obrázek 3.11: *Predikcia textu*

- JQuery algoritmus - Vstupný algoritmus získava zadané písmená alebo slová z formulára príkazového riadku v reálnom čase. Získane dáta zasiela do spracovacieho mechanizmu. Na druhej strane je tu výstupný algoritmus, ktorý okamžite zobrazí výstupne informácie vyhladané v databáze. Skript sa nachádza v priečinku /script a názov súboru je script.js.
- Spracovanie vstupných dát - Je to algoritmus, ktorý vytvorí požiadavku z prijatých dát a nasledovne ju zašle na spracovanie do databázy. Tento algoritmus je vytvorený pomocou PHP ktorý je umiestnený v priečinku /script a názov súboru je commandline.php.
- Spracovanie výstupných dát - Výstupne dáta z databázy tento mechanizmus upraví do podoby, ktorá je vhodná pre zobrazenie, ktoré má formu predikcie textu. Tento mechanizmus sa nachádza v súbore commandline.php.

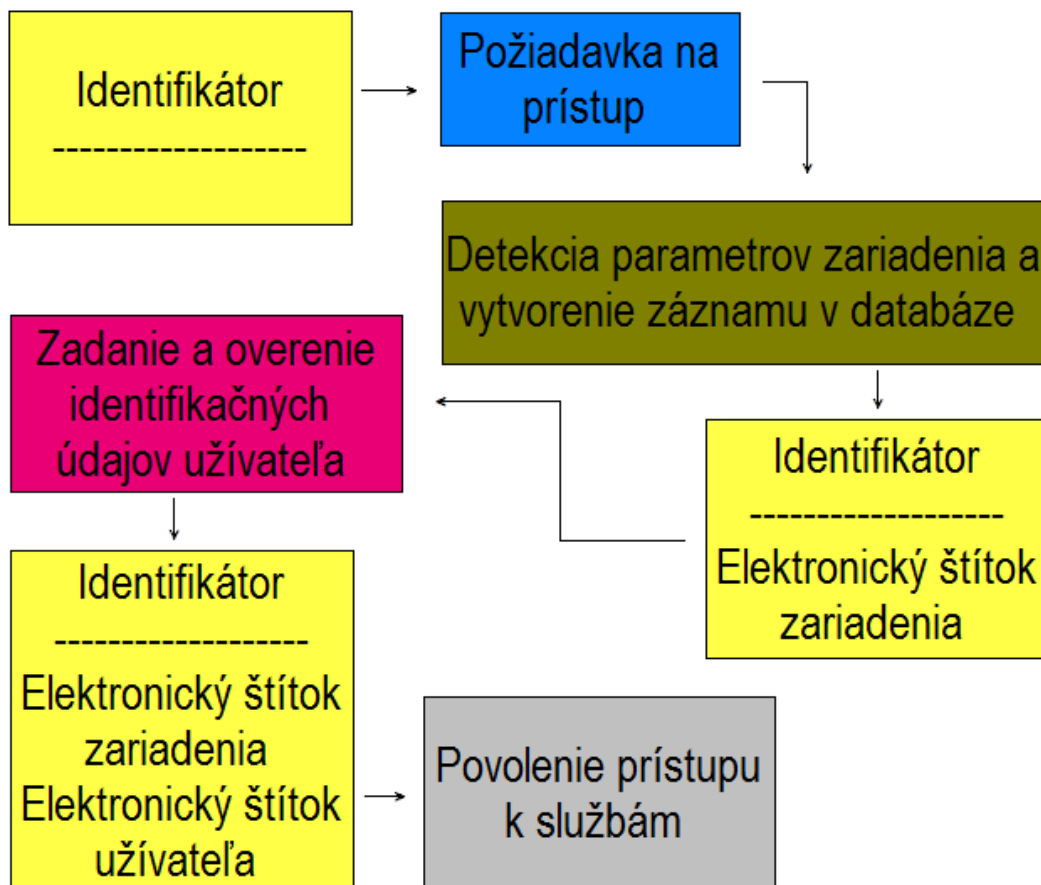
Momentálne je implementovaný iba mechanizmus vyhľadávania informácií. V budúcnosti by mal pribudnúť mechanizmus, ktorý vykonáva požadované operácie pomocou kľúčových slov ako je napríklad rýchla tvorba textových správ.

### 3.3.3 Prezencia

V bežne dostupných systémoch sa pre autentizáciu pridáva užívateľovi iba jeden identifikačný elektronický štítok. Ak užívateľ prístupuje z viacerých zariadení súčasne, je tento identifikačný štítok spoločný pre všetky zariadenia, má to tú nevýhodu, že systém nedokáže rozlíšiť, ktoré zariadenie je momentálne aktívne. Pre naše potreby je tento jednoduchý mechanizmus nedostatočný, preto sme nútený používať viac úroveň autentifikáciu. Náš systém



bude prideloť minimálne dva typy identifikačných štítkov. Prvý z nich je identifikátor zariadenia, ktorý zostane pridelený v momente prvého prístupu k službe. Tento štítok nesie informáciu o type zariadenia, o čase kedy si zariadenie vymieňalo informácie so systémom a rôzne ďalšie informácie. Po identifikácii užívateľa menom a heslo systém prideli druhý identifikačný štítok ktorý nesie základné informácie o užívateľovi. Pomocou tých identifikátor nám môže systém sprístupniť svoje služby.



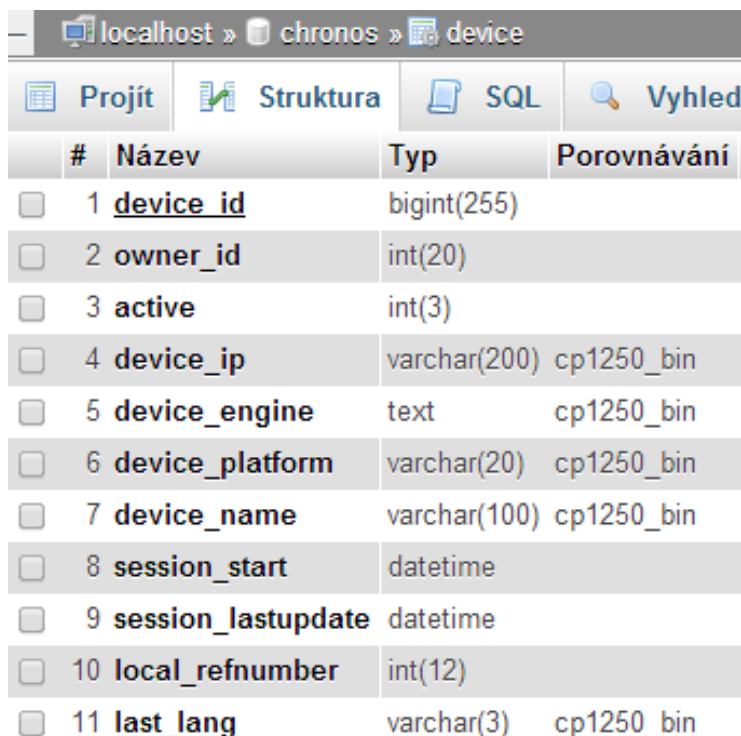
Obrázek 3.12: Kroky potrebné k povoleniu prístupu k službám

Pre pridelenie identifikátora musí nasledovať proces, pri ktorom si systém overí a získa potrebné informácie. Postupnosť týchto krokov je zobrazená na obrázku 3.12. Údaje o týchto identifikátoroch sú uschované v databáze po určitú dobu, pre účely spätnej kontroly a získavania informácií o type prístupových zariadení. Vytvorené kódy k tomuto mechanizmu sú umiestnené v knižnici features.php. Ďalšia časť mechanizmu pre autentizáciu sa nachádza v súbore login.php ktorý prideli identifikačnú značku užívateľa. Pre nás sú najpodstatnejšie tieto funkcie:

- add\_session() - Táto funkcia priradiť identifikačný štítok jednotlivým zariadeniam. Najprv si overí, či prístupové zariadenie už niekedy v minulosti pristupovalo na užívateľské rozhranie, ak má informácie o tomto zariadení, tak len priradi

identifikačný štítok, ak nemá tak najprv vytvorí záznam v databáze a potom prideli identifikačný štítok zariadenia.

- `insert_mysql_device()` - Je to funkcia, ktorá získa potrebné informácie o zariadení a vytvorí potrebný dotaz, ktorý nasledovne pošle do databázy na spracovanie. Ak bol dotaz spracovaný, tak systém povolí pridelenie identifikačného štítku.



	#	Název	Typ	Porovnávaní
<input type="checkbox"/>	1	<u>device_id</u>	bigint(255)	
<input type="checkbox"/>	2	owner_id	int(20)	
<input type="checkbox"/>	3	active	int(3)	
<input type="checkbox"/>	4	device_ip	varchar(200)	cp1250_bin
<input type="checkbox"/>	5	device_engine	text	cp1250_bin
<input type="checkbox"/>	6	device_platform	varchar(20)	cp1250_bin
<input type="checkbox"/>	7	device_name	varchar(100)	cp1250_bin
<input type="checkbox"/>	8	session_start	datetime	
<input type="checkbox"/>	9	session_lastupdate	datetime	
<input type="checkbox"/>	10	local_refnumber	int(12)	
<input type="checkbox"/>	11	last_lang	varchar(3)	cp1250_bin

Obrázek 3.13: Štruktúra tabuľky device

Elektronický štítok zariadenia je previazaný s konkrétnym záznamom v databáze, spoločne tvoria pár, ktorý sa pravidelne synchronizuje. Elektronický štítok nesie v sebe iba základné informácie potrebné pre základnú identifikáciu. Ma to výhodu v bezpečnosti lebo citlivé údaje spracovávame vo vnútri systému a neukladajú sa v cache pamätiach jednotlivých prehliadačov internetu. Štruktúru tabuľky môžeme vidieť na obrázku 3.13. Potrebné informácie ktoré systém potrebuje sú:

- `device_id` - Jedinečné identifikačné číslo zariadenia.
- `owner_id` - Identifikačné číslo užívateľa, ak je užívateľ anonymný, pretože konkrétne zariadenie žiada prístup k službe prvý krát ma tato položka hodnotu 0.
- `active` - Stav zariadenia, hodnota nula prezentuje neaktívne zariadenie, hodnota 1 aktívne zariadenie a ostatné hodnoty môžu reprezentovať rôzne stavy.
- `device_ip` - IP adresa zariadenia.
- `device_engine` - Použitý internetový prehliadač.
- `device_platform` - Označuje typ zariadenia.
- `device_name` - Užívateľom pridelené meno zariadenia.
- `session_start` - Dátum a čas, kedy zariadenie prvý krát pristupovalo k službe.

- session\_lastupdate - Dátum a čas poslednej požiadavky zariadenia.
- local\_refnumber - Lokálne referenčné číslo.
- last\_lang - Posledný preferovaný jazyk.

#### 3.3.3.1 *Detekcia typu zariadenia*

Pre lepšie prispôsobenie poskytovaných služieb, pre rôzne druhy prístupových zariadení, musíme v prvom rade zistiť o aké zariadenia sa jedná a aký operačného systému používa. Trh s komunikačnými zariadeniami je veľmi dynamický a máme tu niekoľko druhov operačných systémov. Preto je najvýhodnejšia forma detekcie pomocou externej aplikácie. V našom systéme je použité otvorene riešenie, ktoré sa nazýva Mobile Detect a spadá pod MIT License (MIT), ktorá nás oprávňuje používať túto aplikáciu pre nekomerčné účely. Hlavnou úlohou tejto aplikácie je určenie o aké zariadenia sa jedná, určí či prístupové zariadenie je osobný počítač, tablet alebo inteligentný telefón. Knižnica tejto aplikácie sa nachádza v priečinku /script a názov súboru je Mobile\_Detect.php. Pre spustenie detekcie zariadenia použijeme tento zdrojový kód.

```
$detect = new Mobile_Detect;  
  
$deviceType = ($detect->isMobile() ? ($detect->isTablet() ?  
'tablet' : 'phone') : 'computer');
```

#### 3.3.3.2 *Detekcia prítomnosti užívateľa*

Pre zabezpečenie vyššej kvality smerovania komunikácie je detekcia prítomnosti užívateľa jednou z dôležitých informácií. Ak je užívateľ prihlásený na svojom užívateľskom účte to hneť nezmámená že je tu fyzický prítomný. Úlohou tohoto mechanizmu je určenie či je užívateľ fyzický prítomný na konkrétnom prístupovom zariadení. Mame dve možnosti ako to zistiť, jedna z nich je že užívateľ sám ukončí komunikáciu pomocou zariadenia ak používa viac zariadení súčasne. Druhá možnosť je definovanie časového intervalu od poslednej požiadavky na poskytnutie služby, pre prekročení tohoto intervalu systém automaticky vyhodnotí že toto zariadenie užívateľ momentálne toto zariadenie nepoužíva. Každý identifikačný štítok je prepojený zo známom v databáze do ktorého ukladá rôzne informácie, medzi nimi je aj čas poslednej požiadavky. Potrebne kódy funkcií sa nachádzajú v knižnici features.php. Hlavné funkcie potrebné pre funkčnosť tohoto mechanizmu sú :

- update\_mysql\_device\_active() - Slúži na manuálne definovanie stavu zariadenia, tato funkcia upraví stav zariadenia na aktívny. Zvyčajne sa tato funkcia vykonáva pri prihlásení užívateľa do systému.
- update\_mysql\_device\_deactive() - Je presným opakom predchádzajúcej funkcie. Upraví stav zariadenia na neaktívny. Zvyčajne sa vykonáva pri odhlásení užívateľa zo systému.
- update\_mysql\_device\_lastupdate() - Tato funkcia je súčasťou automatického mechanizmu určenia stavu zariadenia. Jej úlohou je úprava časového intervalu v databáze pri každej požiadavke pre poskytnutie určenej služby. Časový údaj sa nachádza v zazname ktorý je prepojený s identifikačným štítkom.

### 3.3.3.3 *Autentizácia*

Autentizáciu užívateľa vykonávame pomocou identifikačných údajov užívateľa. Štandardne sa používa užívateľské meno a heslo. V databáze nie je priamo uložené heslo ale iba odtlačok tohoto hesla pre prípad zneužitia tohoto hesla. Odtlačok je vytvorený pomocou hašovacej funkcie MD5. Pri prístupe užívateľ zadá meno a heslo a po overení mu systém povolí prístup a priradí identifikačnú značku alebo mu prístup zamietne. Zdrojové kódy funkcií sú umiestnené v súbore login.php. Pri odhlásení používateľa mu systém odoberie identifikačnú značku a presmeruje ho na prihlasovací formulár. Zdrojové kódy pre odobratie identifikačnej značky sa nachádzajú v súbore s názvom logoff.php. Funkcia auth\_session() ktorá sa nachádza v knižnici features.php, má za úlohu overovať užívateľské značky, ak užívateľ disponuje potrebnou značkou, systém mu povolí prístup k službám, ak nedisponuje správnou identifikačnou značkou, tak ho systém presmeruje na miesto, kde túto značku po overení môže získať.

### 3.3.3.4 *Autorizácia*

O pridelovanie práv pre používanie rôznych služieb sa stará autorizačná služba. Prideluje autorizačnú značku a pri poskytovaní služieb overuje, či táto značka má oprávnenie pristupovať k oprávnenej službe. Užívateľské oprávnenia sa delia do rôznych skupín, ktoré sú vopred definované. O pridelovanie užívateľov k rôznym skupinám sa stará administrátor systému alebo k tomu oprávnená osoba.

## 3.3.4 **Text To Speech**

TTS je veľmi zložitá operácia z technologického hľadiska. V našom systéme je implementovaný mechanizmus prekladu textu na hlas pomocou služieb, ktoré má vo svojich produktoch integrovaná spoločnosť Google. Úlohou nášho mechanizmu je predpríprava formy vstupných dát a získanie konkrétnej zvukovej stopy, ktorá sa nasledovne uloží na vopred vyhradenom mieste v pamäti. Tento proces musí bežať v pozadí systému a nemôže ovplyvňovať prácu v užívateľskom prostredí. Mechanizmus musí zabezpečiť nielen preklad textových správ, ale aj rôznych funkcií systému, ako je napríklad uvítanie v systéme alebo rôzne hlasové upozornenia.

### 3.3.4.1 *TTS služby poskytnuté produktami Google*

Spoločnosť Google ponúka hlasový modul pre svoj jazykový prekladač. Hlasový modul je integrovaný do tejto služby a pomáha vylepšovať kvalitu poskytovaných produktov. Systém využíva tento hlasový modul pre vlastné použitie. Komunikácia so službou prebieha pomocou URL adresy http protokolu. Pomocou presne definovaného reťazca, v ktorom je umiestnený text, ktorý chceme preložiť na hlas. Po spracovaní našej požiadavky nám modul poskytne zvukovú stopu, ktorú musíme pomocou funkcie uložiť na predom definovaný diskový priestor a upraviť záznam v databáze, ktorý reprezentuje túto zvukovú stopu. Nasledovne môžem túto zvukovú stopu prehrať pomocou zvukového prehrávača integrovaného v užívateľskom rozhraní.

### 3.3.4.2 *Príprava požiadavku na preklad textu do hlasu*

Prvou hlavnou úlohou je príprava vhodnej formy textu tak, aby ju hlasový modul prijal. V knižnici `features.php` sa nachádza funkcia `tts_voice_convert_voice()`, ktorá nám automaticky spracuje všetky požiadavky na spracovanie hlasu. Užívateľské rozhranie po zadaní požiadavky na preklad textu nečaká na vykonanie požiadavky ale vráti sa do požadovaného stavu a nasledovne na pozadí kontroluje či systém požiadavku spracoval, ak ju systém spracoval tak ponúkne možno si túto zvukovú stopu vypočúť. Príprava požiadavku má dva hlavné kroky. Prvý z nich je prekódovanie textu zo znakovkej sady `cp1250` na znakovú sadu `UTF-8`. V druhom kroku je príprava požadovaného URL reťazca, ktorý má nasledovnú podobu.

```
http://translate.google.com/translate_tts?ie=utf-8&q="Vstupný text v UTF-8"&tl="Požadovaný jazyk hlasovej spravi"
```

Pre potreby prekódovania znakovkej sady používa funkciu, ktorá má názov `winutf8()` a nachádza sa v knižnici `features.php`.

### 3.3.4.3 *Získanie a uloženie zvukovej stopy*

Po vytvorení vhodného URL reťazca musíme nasledovne tento reťazec zaslať hlasovému modulu na spracovanie. Po spracovaní nám modul ponúkne možnosť uloženia tejto zvukovej stopy. Odoslanie a uloženie zvukovej stopy vykoná PHP modul `cURL`, ktorý spracováva URL syntax. Nasledujúci algoritmus vykonáva požadované operácie.

```
$filename = md5(date(DATE_RFC2822)).'.mp3';  
$fileurl = 'http://translate.google.com/translate_tts?ie=utf-8&q=' . $encoded . '&tl=' . $lang;  
  
• $filename - Názov pre vytvorenú zvukovú stopu, ktorý musí byť jedinečný. Pomocou  
  hašovacej funkcie vytvoríme takýto názov. Vstupné dáta pre túto funkciu sú dátum a  
  čas, ktorý nám zabezpečí jedinečnosť názvu.  
• $fileurl - Je to pripravený URL reťazec, ktorý posielame hlasovému modulu.  
• $encoded - Je to text v znakovkej sade UTF-8.  
• $lang - Skratka požadovaného jazyka, napríklad sk, cs, en.  
  
$ch = curl_init($fileurl);  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);  
$data = curl_exec($ch);  
curl_close($ch);  
file_put_contents('temp/tts/' . $filename, $data);
```

Nasledovne prebehne odoslanie požiadavku do hlasového modulu. `file_put_contents` nám zabezpečí uloženie zvukovej stopy do priečinka `/temp/tts/`. Názov zvukovej stopy bude vygenerovaný reťazec pomocou `md5`.

#### 3.3.4.4 Spracovanie požiadavku transformácie textu na hlas

Spracovanie požiadaviek pre služby TTS prebieha na pozadí systému v rôznych časových cykloch, pretože spracovanie hlasu môže trvať určitú dobu. Ak by sme nezaviedli tento mechanizmus, tak by mohlo dochádzať k spomaľovaniu reakcií užívateľského rozhrania, pretože by muselo čakať na spracovanie hlasu. Týmto spôsobom systém vloží požiadavku a pracuje ďalej, v určitých časových intervaloch kontroluje momentálny stav požiadavky, ak bola splnená, užívateľ si ju môže prehrať pomocou integrovaného zvukového prehrávača. Algoritmus, ktorý spracováva tieto požiadavky sa nazýva `tts_voice_convert_voice()` a je uložený v knižnici `features.php`. Požiadavka je reprezentovaná pomocou záznamu v tabuľke `tts_voice`, táto tabuľka má štruktúru zobrazenú na obrázku 3.14.

#	Názov	Typ	Porovnávač
<input type="checkbox"/> 1	<u>tts_voice_id</u>	bigint(255)	
<input type="checkbox"/> 2	tts_date	datetime	
<input type="checkbox"/> 3	tts_owner	int(255)	
<input type="checkbox"/> 4	tts_input	text	cp1250_bin
<input type="checkbox"/> 5	voice_tempfile	varchar(255)	cp1250_bin
<input type="checkbox"/> 6	tts_voice_created	tinyint(1)	
<input type="checkbox"/> 7	tts_mess_parents	int(255)	
<input type="checkbox"/> 8	tts_voice_lang	varchar(255)	cp1250_bin

Obrázek 3.14: Štruktúra tabuľky `tts_voice`

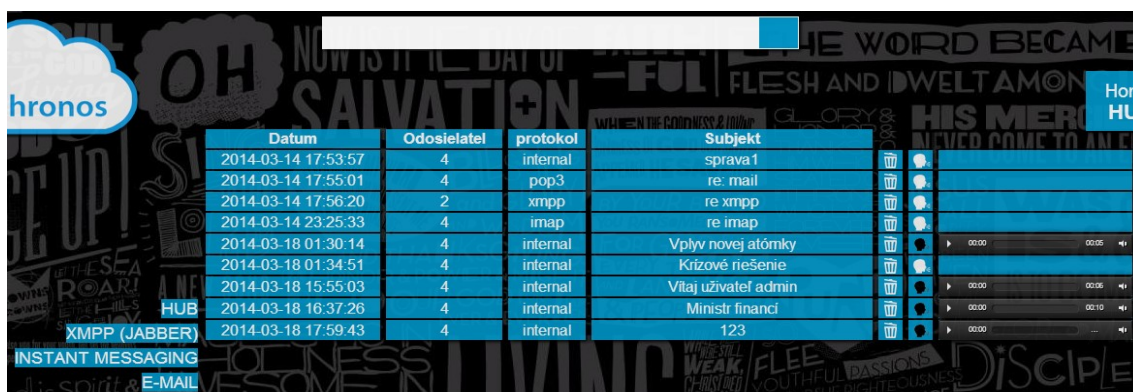
- `tts_voice_id` - Jedinečné identifikačné číslo požiadavky.
- `tts_date` - Dátum a čas vloženia požiadavky.
- `tts_owner` - Identifikačné číslo žiadateľa.
- `tts_input` - Vstupný text správy.
- `voice_tempfile` - Názov vytvorenej zvukovej stopy
- `tts_voice_created` - Stav požiadavky, hodnota 0, požiadavka čaká na spracovanie, hodnota 1, požiadavka je spracovaná.
- `tts_mess_parents` - Ak je prekladaný vstup textová správa, tak v tomto poli sa nachádza identifikačné číslo tejto správy.
- `tts_voice_lang` - Informácia o jazyku vstupného textu.

## 4 Zhodnotenie dosiahnutých výsledkov

Úlohou pri návrhu platformy bolo opísať základne problémy, ktoré rieši zjednotená komunikácia a aké výhody nám môže priniesť integráciu takéhoto produktu do firemnej infraštruktúry. Návrh systému je rozdelený do teoretickej časti a praktickej časti. V teoretickej časti sme načrtli ako by mohla vyzeráť platforma z technologického hľadiska. Systém, ako celok sme rozdelili do hlavných blokov, ktoré sme detailnejšie analyzovali. Analýza nám ukázala s akým typom dát každý blok pracuje, aká je vnútorná štruktúra modulu a jednotlivých blokov a akým spôsobom pracujú jednotlivé mechanizmy a funkcie. V praktickej časti sme teoretický návrh pretavili do reálnej implementácie rôznych častí systému. Hlavne pomocou skriptovacieho jazyka PHP a rôznych ďalších programovacích jazykov sme navrhli funkcie, algoritmy a mechanizmy, ktoré získavajú, spracúvajú a ukladajú potrebné informácie a údaje. Vytvorili sme webové užívateľské rozhranie, podľa zadaných požiadaviek, do ktorého sme integrovali mechanizmy autentizácie, autorizácie a ďalšie funkcie detekcie zariadení pre správne fungovanie systému. Pracovné prostredie webového užívateľského rozhrania je intuitívne a nasleduje najnovšie trendy v oblasti užívateľského komfortu a grafického spracovania. V pracovnom prostredí sme hlavne integrovali spracovanie textu na hlas, ktorý má byť jednou z hlavných poskytovaných služieb. Ďalším prvkom je príkazový riadok, ktorý má dôležitú funkciu pre zrýchlenie práce s týmto UC systémom. Pomocou tohto príkazového riadku môže rýchlo a jednoducho vyhľadávať informácie a vykonávať potrebné úkony za pomocou kľúčových slov. Predikcia textu urýchlí celkovú schopnosť získavania a spracovania informácií. Všetky vytvorené kódy a skripty budú priložené vo forme prílohy k tejto práci.

### 4.1 Ďalšie časti systému

Zjednotené zobrazenie všetkých prijatých správ sa nazýva hub. Je to časť systému kde sa zobrazujú všetky správy spoločne alebo ich môžeme roztriediť podľa komunikačných protokolov, ku ktorým patria. Jednotlivé správy môžeme vymazať alebo dať pokyn na prevod správy na hlas. Po spracovaní si môžeme túto zvukovú stopu vypočítať pomocou integrovaného prehrávača. Potrebné funkcie a mechanizmy nájdeme v súbore index.php a v knižniciach call\_method.php a call\_iframe.php. Ukážka tohoto prostredia sa nachádza na obrázku 4.1.



Datum	Odosielateľ	protokol	Subjekt
2014-03-14 17:53:57	4	internal	sprava1
2014-03-14 17:55:01	4	pop3	re. mail
2014-03-14 17:56:20	2	xmpp	re xmpp
2014-03-14 23:25:33	4	imap	re imap
2014-03-18 01:30:14	4	internal	Vplyv novej atómy
2014-03-18 01:34:51	4	internal	Krízové riešenie
2014-03-18 15:55:03	4	internal	Vitaj užívateľ admin
2014-03-18 16:37:26	4	internal	Ministr financií
2014-03-18 17:59:43	4	internal	123

Obrázek 4.1: Ukážka zjednoteného zobrazenia správ





Obrázek 4.2: Ukážka prihlasovacieho formulára

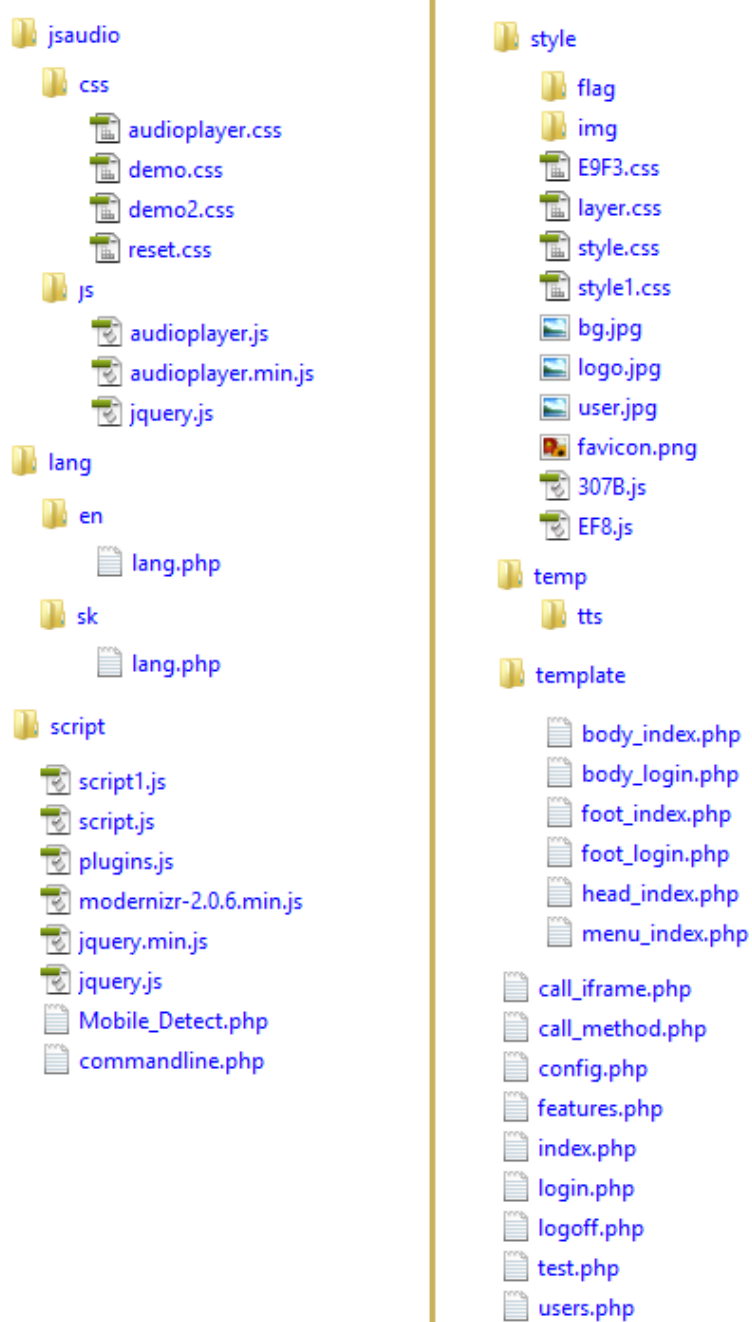
Pri vstupe na webové rozhranie platformy sa nám zobrazí prihlasovací formulár, ktorý je zobrazený na obrázku 4.2. Pomocou tohoto formulára zadáme prihlasovacie údaje užívateľa a po overení nám systém povolí prístup. Pod formulárom sa nachádzajú prvky pomocou ktorých môžeme zmeniť zobrazovaný jazyk. V súbore z názvom login.php najedene potrebné funkcie k zobrazeniu tohto formulára.

## 4.2 Stromová architektúra priečinkov a pod priečinkov

Rozdelenie súborov do jednotlivých priečinkov má presne definované usporiadanie podľa funkcie každého súboru. Toto usporiadanie vyplýva z požiadaviek modularity systému, môžeme ho vidieť na obrázku č.4.3. Pre systém sú najdôležitejšie priečinky:

- Lang - Priečinok ktorý uchováva súbory potrebné pre podporu jazyka.
- Script - Nachádzajú sa tu pomocné funkcie, ktoré sú napísané pomocou iného skriptovacieho jazyka, napríklad Javascript alebo perl.
- Style - Priečinok, kde sa nachádzajú grafické prvky rozhrania a súbory kaskádových štýlov.
- Temp - Dočasný ukladací priestor pre vytvorené súbory ako sú zvukové stopy.
- Template - Priečinok, ktorý uchováva súbory v ktorých sa nachádza štruktúra webového užívateľského rozhrania.
- Koreňový priečinok - Je to základný priečinok. Hlavné súbory sa nachádzajú priamo v tomto priečinku a nie v ďalších pod priečinkoch.





Obrázek 4.3: *Stromová architektúra priečinkov a súborov*

## Záver

Účelom diplomovej práce je implementácia zjednotenej komunikácie v podnikovom prostredí. V prvej kapitole sa zaoberáme problematikou, ktoré vznikajú pri používaní rôznych komunikačných kanálov a o nástrahách ako sú súkromná komunikácia pre firemne účely, nebezpečenstvo uniku informácii a nedokumentovaná komunikácia. V druhej kapitole sme sa pozreli bližšie na dostupné protokoly, ktoré používajú komunikačné kanály ako elektronická pošta, Instant Messaging a VoIP. Ďalej sme uviedli, ktoré komponenty budeme využívať pri implementácii platformy. Tretia kapitola sa zaoberá teoretickým a praktickým návrhom realizácie platformy. V teoretickom návrhu sú uvedené požadované vlastnosti platformy pre mobilitu a flexibilitu systému, jazykovú podporu, modularitu systému, spracovanie hlasu TTS, užívateľské rozhranie, manažment zdrojov, bezpečnosť komunikácie a firemných správ. Navrhli sme koncepciu platformy, ktorú sme rozdelili do jednotlivých modulov. Modul jadro nám zabezpečuje základné funkcie pre fungovanie systému. Overuje identitu užívateľa a povoľujú prístup k službám. IM a poštový modul integruje komunikačné protokoly do systému tak, aby sme pomocou tohto modulu mohli prijímať a odosielať správy. Hlasový modul poskytuje TTS služby pre preklad textu na hlas. Telefónny modul integruje pobočkovú ústredňu asterisk do systému. Pre reálne používanie služieb tu máme užívateľské rozhranie. A o najrýchlejšie doručenie správ sa bude starať modul smerovania komunikácie. V praktickej časti implementujeme teoretické návrhy platformy do rôznych funkcií, mechanizmov a aplikácií. Pomocou skriptovacieho jazyka sme vytvorili algoritmy potrebné pre funkčnosť systému ktoré sú uložené v jadre systému. Vytvorili sme návrh grafického webového rozhrania podľa zadaných požiadaviek na rozhranie. Do tohto rozhrania sme implementovali mechanizmy autentizácie a autorizácie, ktoré nám overujú identitu užívateľa a pridelujú mu prístupové práva. Do webového rozhrania sme ďalej implementovali pracovné prostredie, ktoré nasleduje najnovšie trendy v oblasti užívateľských rozhraní. Pracovné prostredie sme rozdelili do troch častí, jednou z nich je aj zjednotené zobrazenie prijatých správ. Tieto správy môžeme pomocou navrhnutého TTS modulu preložiť na hlas. Túto zvukovú stopu si môžeme po spracovaní vypočuť na mieste. Preklad textu na hlas nám zabezpečuje externý hlasový modul od spoločnosti Google. V poslednej kapitole sme zhrnuli dosiahnuté výsledky, opísali adresárovú štruktúru priečinkov. Všetky vytvorené zdrojové kódy, obsah a štruktúra databázy sú poskytnuté vo forme prílohy k tejto práci. Na vytvorený systém sa vzťahuje licencia GNU GPL a to znamená že je voľne šíriteľný.

Výhody UC systému sú nepopierateľné, pretože má riešenie na množstvo problémov, ktoré vznikajú pri komunikácii. Preto, aby sa táto technológia presadila na komunikačnom trhu, musí prísť produkt, ktorý bude integrovať väčšinu komunikačných kanálov a podporovať telefónne pobočkové ústredne od rôznych výrobcov.

---

## Použitá literatura

- [1] ASTERISK. *Asterisk* [online]. 2013 [cit. 2013-07-07]. Dostupné z: <http://www.asterisk.org/>
- [2] Mobile Detect. *Mobile Detect* [online]. [cit. 2014-05-05]. Dostupné z: <http://mobiledetect.net>
- [3] M. Vozňák, Voice over IP. Vysokoškolská skripta, Vydavatel: VŠB-TU Ostrava, Dotisk prvního vydání, v Ostravě, 2009, ISBN 978-80-248-1828-3
- [4] Post Office Protocol - Version 3. *Post Office Protocol - Version 3* [online]. 1996 [cit. 2014-05-06]. Dostupné z: <http://www.ietf.org/rfc/rfc1939.txt>
- [5] INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1. *INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1* [online]. 2003 [cit. 2014-05-06]. Dostupné z: <http://tools.ietf.org/html/rfc3501d>
- [6] SIMPLE MAIL TRANSFER PROTOCOL. *SIMPLE MAIL TRANSFER PROTOCOL* [online]. 1982 [cit. 2014-05-06]. Dostupné z: <http://tools.ietf.org/html/rfc821>
- [7] Extensible Messaging and Presence Protocol (XMPP). *Extensible Messaging and Presence Protocol (XMPP)* [online]. 2011 [cit. 2014-05-06]. Dostupné z: <https://tools.ietf.org/html/rfc6120>
- [8] Integrace VoiceXML do Asterisku. *Integrace VoiceXML do Asterisku* [online]. 2013 [cit. 2014-05-06]. Dostupné z: <https://dspace.vsb.cz/handle/10084/98629>
- [9] PHPAGI. *PHPAGI* [online]. 2004-2010. [cit. 2014-05-06]. Dostupné z: <http://phpagi.sourceforge.net>
- [10] M. Voznak, K. Tomala, J. Vychodil, J. Slachta, Advanced concept of voice communication server on embedded platform, *Przeglad Elektrotechniczny*, Volume 89, Issue 2 B, 2013, pp. 228-233.
- [11] L. Macura, M. Voznak, J. Slachta, Unified Administration of VoIP Communication Systems, In Proc. International Conference on Telecommunication Systems, Modeling and Analysis (ICTSM2012), Prague, May 24-26, 2012, pp. 188-197, ISBN 978-0-9820958-6-7.

---

## Zoznam príloh

Súčasťou DP je CD.

Adresárová štruktúra priloženého CD:

- ./Databaza
- ./Rozhranie
- ./Rozhranie/jsaudio
- ./Rozhranie/jsaudio/css
- ./Rozhranie/jsaudio/ls
- ./Rozhranie/lang
- ./Rozhranie/lang/en
- ./Rozhranie/lang/sk
- ./Rozhranie/script
- ./Rozhranie/style
- ./Rozhranie/style/flag
- ./Rozhranie/style/img
- ./Rozhranie/temp
- ./Rozhranie/temp/tts
- ./Rozhranie/template